University of California
Los Angeles

Stability and Performance of Adaptive Filters without Slow Adaptation Approximations

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical Engineering

by

Vítor Heloiz Nascimento

1999

The dissertation of Vítor Heloiz Nascimento is approved.

Fernando Paganini

D. Lewis Mingori

Alan Willson

Ali H. Sayed, Committee Chair

University of California, Los Angeles

1999

To my wife Naomi, with deepest love and gratitude.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank first a friend and mentor whose help and support was fundamental to me. Unfortunately, Prof. Normonds Alens passed away during my stay at UCLA, but he will be always remembered, for his friendship and his joy for life and for research.

I would also like to thank my M.S. advisor, Prof. Luís de Queiroz Orsini, for his continuing support and interest in my progress; and my many friends at the University of São Paulo, who helped in numerous ways to make my studies at UCLA possible.

It is difficult to express my gratitude to my advisor, Prof. Ali H. Sayed, for his orientation and advice during these years, for his careful review of all my work, his insightful comments and suggestions, and his commitment to clear presentations.

My gratitude also goes to all the professors, both at UCLA and at UCSB, who shared with me their views, knowledge, and experience, and helped me with their comments and discussions. I am also indebted to my fellow students at UCSB and UCLA, for the discussions and for the sometimes fun, sometimes hard time we had together, and to Dr. Patrick Williams, from Litton Data Systems.

I am indebted to the people from CNPq, in special to Alcina T. Queiroz, for their attention and support.

I would like to express my deepest gratitude to my parents Benedicto and Magdalena, and to my brother Carlos, for their encouragement and help from far away.

I also cannot thank enough my wife Naomi Nascimento, for her confidence, love, and continuous support at all moments. Not contented with that, she also helped me by writing simulation programs in C++, and in the solution of a few

problems in some very important moments. I am indeed very fortunate to have her friendship and love.

This work was supported by a Fellowship from the *Conselho Nacional de Pesquisa e Desenvolvimento* — CNPq, Brazil, and by the *Escola Politécnica*, *Universidade de São Paulo*, Brazil. It was also supported by the National Science Foundation under Award numbers MIP-9409319, MIP-9796147, and CCR-9732376.

Vita

April 22, 1967	Born in São Paulo, Brazil.
1987–1989	Undergraduate student researcher, University of São Paulo, Brazil.
1988	*Prêmio Gastão Vidigal*, awarded to the student with the highest scores in the course *General Economics*, School of Engineering, University of São Paulo.
1989	B.S., Electrical Engineering, University of São Paulo.
1990	*Prêmio Armando Álvares Penteado*, awarded to the top three graduates of the School of Engineering, University of São Paulo.
1990–1994	Instructor, School of Engineering, University of São Paulo.
1992	M.S., Electrical Engineering, University of São Paulo.
1993–1994	Co-Coordinator of the undergratuate *Laboratory of Electric Measurements*, School of Engineering, University of São Paulo.

Abstract of the Dissertation

Stability and Performance of Adaptive Filters without Slow Adaptation Approximations

by

Vítor Heloiz Nascimento

Doctor of Philosophy in Electrical Engineering

University of California, Los Angeles, 1999

Professor Ali H. Sayed, Chair

The performance of an adaptive filter is crucially dependent on its rate of convergence, steady-state mean-square error, and stability properties, especially in finite-precision implementations. Exact performance analyses only exist for infinitesimally small step-sizes or under certain so-called independence assumptions. There are practically no counterparts of these analyses for larger step-sizes in the literature. Such results are desirable since they would serve as a guide for the design, and also for a better understanding, of adaptive filters with faster convergence speeds. Progress in this direction is often hindered by the complexity of the (possibly time-variant and nonlinear) update relations that arise when slow adaptation approximations are not employed.

This dissertation develops techniques for the stability and performance analyses of adaptive filters without resorting to slow adaptation approximations. The work expands the four main methods of analysis that have been used so far in the literature, namely, mean-square stability analysis, almost-sure stability analysis, Lyapunov stability analysis, and analysis by simulation or experimentation.

Among the original contributions of this work are the first computable lower

bound on the largest step-size that guarantees mean-square stability in the absence of the independence assumptions (Ch. 4); a detailed study of the behavior and properties of ensemble-average learning curves and how special care is needed in using them to predict or evaluate the performance of an adaptive filter (Ch. 5); a proof that an adaptive filter can actually have two rates of convergence; one rate for the initial phase of operation and another faster rate for later time instants (Ch. 5); a new leakage-based algorithm that avoids both the drift and bias problems of existing adaptive methods (Ch. 6); and a Lyapunov stability analysis for floating-point implementations in worst-case scenarios (Ch. 6). Further contributions to the independence theory itself are provided in Chs. 2 and 3, especially an analysis of the normalized LMS algorithm. The introduction and the concluding remarks of each chapter indicate the specific contributions of that chapter and how they relate to available results in the literature.

CHAPTER 1

MOTIVATION AND CONTRIBUTIONS

In this chapter we provide an overview of the class of stochastic-gradient adaptive algorithms that we shall deal with in this dissertation. In particular, we review the least-mean squares (LMS) algorithm, the normalized LMS algorithm, and the leaky LMS algorithm. We also comment briefly on the common framework for the analysis of these algorithms via the independence assumptions, and we end the chapter by explaining the scope and contributions of this work.

1.1 LINEAR ESTIMATION

The problem of *linear least-mean squares estimation* is the following. Given a zero-mean scalar sequence of random variables $\{y(k)\}_{k=0}^{\infty}$ and another sequence of zero-mean length-M random vectors $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$, we seek length-$M$ vectors $\{\boldsymbol{w}_{*,k}\}$ such that

$$\hat{y}(k) = \boldsymbol{x}_k^T \boldsymbol{w}_{*,k}$$

is the optimal linear estimate for $y(k)$, in the sense that $\boldsymbol{w}_{*,k}$ solves

$$\sigma_v^2(k) \stackrel{\Delta}{=} \min_{\boldsymbol{w}} \mathrm{E}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}\big)^2,$$

where E is the expectation operator and ^{T} is the transposition operator. Define

$$v(k) \stackrel{\Delta}{=} y(k) - \hat{y}(k) = y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_{*,k}.$$

The sequence $\{y(k)\}$ is called the *desired* sequence and $\{\boldsymbol{x}_k\}$ is the *input* or *regressor* sequence. The subscript k denotes time. The sequence $\{v(k)\}$ is the *noise* (or error) sequence.

If the second-order statistics of $\boldsymbol{x}_k$ and $y(k)$ are known, the weight vector $\boldsymbol{w}_{*,\,k}$ can be determined using the well-known *orthogonality principle* of linear estimation [KSH99]. The principle states that the estimation error $v(k)$ must be uncorrelated with the regressors $\boldsymbol{x}_k$, viz.,

$$\mathrm{E}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_{*,\,k}\big)\boldsymbol{x}_k = \boldsymbol{0}. \tag{1.1}$$

Thus define the autocorrelation matrix of $\boldsymbol{x}_k$

$$R_k \triangleq \Big(\mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T\Big), \tag{1.2}$$

and the cross-correlation vector

$$\boldsymbol{p}_k \triangleq \Big(\mathrm{E}\,y(k)\boldsymbol{x}_k\Big). \tag{1.3}$$

Assume further that

R-1. *The matrix R_k is invertible for all $k \geq 0$.*

Then (1.1) can be solved for $\boldsymbol{w}_{*,\,k}$, resulting in

$$\boldsymbol{w}_{*,\,k} = R_k^{-1}\boldsymbol{p}_k. \tag{1.4}$$

When the desired and input sequences $\{y(k),\ \boldsymbol{x}_k\}$ are jointly wide-sense stationary, the solution does not depend on time, and the above quantities reduce to

$$\boldsymbol{w}_{*,\,k} \equiv \boldsymbol{w}_*, \qquad R_k \equiv R, \qquad \boldsymbol{p}_k \equiv \boldsymbol{p}, \qquad \sigma_v^2(k) \equiv \sigma_v^2.$$

That is, (1.4) reduces to

$$\boldsymbol{w}_* = R^{-1}\boldsymbol{p}. \tag{1.5}$$

This so-called *stationary* case will be the main focus of this dissertation.

An important special instance is when $\{y(k)\}$ and $\{\boldsymbol{x}_k\}$ are Gaussian. In this case, the orthogonality principle also implies that the noise sequence $\{v(k)\}$ is *independent* of the input sequence $\{\boldsymbol{x}_k\}$.

Remark. If the sequences $\{y(k), \boldsymbol{x}_k\}$ happen to be related through a linear model of the form

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}^o + v(k),$$

for some unknown $\boldsymbol{w}^o$, and with a noise sequence $v(k)$ that is uncorrelated with $\boldsymbol{x}_k$, then it can be verified that

$$\boldsymbol{w}_* = \boldsymbol{w}^o.$$

That is, the linear least-mean squares estimator $\boldsymbol{w}_*$ of (1.5) coincides with $\boldsymbol{w}^o$. Hence, whenever we assume that $\{y(k), \boldsymbol{x}_k\}$ are related linearly, we can assume that the coefficient vector relating $y(k)$ and $\boldsymbol{x}_k$ is simply $\boldsymbol{w}_*$.

$\diamondsuit$

1.2 STOCHASTIC-GRADIENT ADAPTIVE ALGORITHMS

In practice, the statistics of the input and desired sequences $\{y(k), \boldsymbol{x}_k\}$ are often unknown (or, at most, only approximations for R and $\boldsymbol{p}$ may be available), so

that the computation of the optimal filter using (1.5) becomes impossible or innacurate. In the time-variant case (1.4), even if R_k and $\boldsymbol{p}_k$ were known exactly, computing (1.4) at every time step can be costly.

The explicit solution of (1.5) can be avoided by using an adaptive algorithm that computes, at each time instant k, a new approximation $\boldsymbol{w}_k$ to $\boldsymbol{w}_*$. Among the most commonly used adaptive algorithms are the *least-mean squares algorithm* or LMS [WS85] and its variants, including *normalized* LMS (NLMS) [TF88], *leaky* LMS [GMW82], *least-mean fourth* (LMF) [WW84], *signed*-LMS and *signed-regressor*-LMS [Ewe94, Ewe97], and some other versions of LMS [Set92, SSB96, TC96]. There are also other important algorithms, in special the *recursive least-squares algorithm* (RLS) [SK94, Hay96], and the *constant modulus algorithm* (CMA) [God80] (used for blind equalization). This dissertation focuses on the LMS algorithm and two of its most distinguished variants: leaky LMS and NLMS.

The main advantages of the LMS algorithm, shared by many of its variants, are its small computational cost, its robustness to modeling errors (such as nonlinearities and quantization errors) [HSK96, RS96, SR98], and its good tracking of non-stationary signals [Ewe94]. One disadvantage is that the algorithm has slower convergence than, for example, the RLS algorithm [Hay96].

1.2.1 The LMS Algorithm

The LMS algorithm computes successive approximations $\boldsymbol{w}_k$ through the recurrence relation

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \boldsymbol{x}_k \big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k\big), \quad \text{with initial condition } \boldsymbol{w}_0. \tag{1.6}$$

The design parameter μ is known as the *step-size*. Two important measures of performance for the LMS algorithm are the errors

$$\tilde{\boldsymbol{w}}_k \stackrel{\Delta}{=} \boldsymbol{w}_* - \boldsymbol{w}_k, \tag{1.7}$$

$$e(k) \stackrel{\Delta}{=} y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k. \tag{1.8}$$

Ideally, we would require the algorithm to reduce $e(k)^2$ from its initial value, and to keep $\mathrm{E}\, e(k)^2$ close to σ_v^2 in steady-state (and $\tilde{\boldsymbol{w}}_k$ close to $\mathbf{0}$). The variance $\mathrm{E}\, e(k)^2$ is known as the *mean-square error*, or *MSE* for short. Similarly, the trace of the covariance matrix of $\tilde{\boldsymbol{w}}_k$, $\mathrm{Tr}(\mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T)$, is known as the *mean-square deviation*, or *MSD*.

From the recursion (1.6) one can deduce that the rate of convergence of the algorithm is greatly affected by the step-size. A very small step-size ($\mu \approx 0$) implies that the weights change very slowly at each iteration, and consequently the convergence rate is small. This is inconvenient, since in many applications the filter output cannot be used before the MSE, $\mathrm{E}\, e(k)^2$, achieves a sufficiently low level. Moreover, in non-stationary environments, slow adaptation may not allow the algorithm to properly track time-variations.

1.2.2 The Leaky LMS Algorithm

The leaky LMS algorithm is used in situations where the covariance matrix R is not guaranteed to be invertible, or is ill-conditioned. In these cases, the LMS algorithm may produce unbounded (or very large) estimates $\boldsymbol{w}_k$. This can cause overflow in finite-precision implementations, which considerably degrades the performance of the algorithm. See [Zah73, Ung74, GMW82, SLJ86, Set92, NS96, BS97] for examples in equalization, and [IK84, IT86, IS96] for the use of leaky LMS in adaptive control. We shall discuss these issues in Chapter 6.

The leaky LMS recursion is given by

$$\boldsymbol{w}_{k+1}^{l} = (1 - \mu\alpha_0)\boldsymbol{w}_k^{l} + \mu\boldsymbol{x}_k\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k\big), \quad \text{with initial condition } \boldsymbol{w}_0, \quad (1.9)$$

where we use the symbol $\boldsymbol{w}_k^l$ to differentiate the leaky LMS weight estimate from the LMS estimate, $\boldsymbol{w}_k$. The constant α_0 is called the *leakage constant*. It can be shown that, while the LMS algorithm attempts to minimize $\mathrm{E}\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}\big)^2$, the leaky LMS algorithm attempts to minimize [Hay96]

$$\mathrm{E}\Big\{\alpha_0\|\boldsymbol{w}\|^2 + \big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}\big)^2\Big\}.$$

The norm $\|\boldsymbol{w}\|$ is the Euclidean norm. The leaky LMS algorithm has two main disadvantages: it introduces a bias in the solution (see Chapter 2), and has a higher computational cost than LMS (see Chapter 6).

1.2.3 The Normalized LMS Algorithm

This algorithm tries to improve the convergence speed of LMS, by using the weight vector $\boldsymbol{w}_k^n$ that solves the following constrained minimization problem:

$$\min_{\boldsymbol{w}_{k+1}^n} \|\boldsymbol{w}_{k+1}^n - \boldsymbol{w}_k^n\|^2 \quad \text{subject to} \quad \big(\boldsymbol{x}_k^T\boldsymbol{w}_{k+1}^n\big) = 0.$$

The solution for the optimization is

$$\boldsymbol{w}_{k+1}^n = \boldsymbol{w}_k^n + \frac{1}{\|\boldsymbol{x}_k\|^2}\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k^n\big).$$

In order to avoid possible divisions by zero, the algorithm is usually modified as follows:

$$\boldsymbol{w}_{k+1}^n = \boldsymbol{w}_k^n + \frac{\mu}{a + \|\boldsymbol{x}_k\|^2}\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k^n\big), \quad (1.10)$$

where $a > 0$ is a constant, and μ is a step-size. It can be shown that this modified algorithm solves the following optimization problem [Say97]:

$$\min_{\boldsymbol{w}_{k+1}^n} \|\boldsymbol{w}_{k+1}^n - \boldsymbol{w}_k^n\|^2 \quad \text{subject to} \quad \big(\boldsymbol{x}_k^T\boldsymbol{w}_{k+1}^n\big) = \left[1 - \frac{\mu\|\boldsymbol{x}_k\|^2}{a + \|\boldsymbol{x}_k\|^2}\right]\big(\boldsymbol{x}_k^T\boldsymbol{w}_k^n\big).$$

The primary disadvantage of the NLMS algorithm is that its computational cost is higher than LMS (there is a division and the computation of a norm in addition to the cost of the basic LMS algorithm).

1.3 THE INDEPENDENCE ASSUMPTIONS

Due to the large range of applications in which the LMS algorithm and its variants have proven successful, there has been a vast literature on the analysis of the performance of these algorithms. Unfortunately, the analysis of such algorithms is not a simple task, unless several simplifying assumptions are made. This is because adaptive filters are often time-variant and nonlinear systems.

The most important of these simplifying conditions are known collectively as the *independence assumptions*. Many of the earlier works (and several recent ones) were based on the independence assumptions [WMG67, WML76, CL84, FW85, Ale87, TF88, Slo93, MA97]. Basically, one assumes the following.

M-1 (Modeling assumption). *The sequences* $\{y(k), \boldsymbol{x}_k\}$ *are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k) \tag{1.11}$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2*, and uncorrelated with* $\boldsymbol{x}_k$*, i.e.,*

$$\mathrm{E}\, v(k)\boldsymbol{x}_k = \mathbf{0}.$$

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-3. *The noise sequence* $\{v(k)\}$ *is independent of the input sequence.*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

In applications, these assumptions are seldom satisfied. For example, in channel equalization, the vectors $\boldsymbol{x}_k$ are formed from a delay line, thus $\boldsymbol{x}_n$ shares all but one of the elements of $\boldsymbol{x}_{n+1}$; $\boldsymbol{x}_n$ and $\boldsymbol{x}_{n+1}$ are clearly not independent. Also, $v(k)$ need not be independent of the regressor sequence $\boldsymbol{x}_k$. This happens, for example, when $v(k)$ also accounts for unmodeled or undermodeled dynamics.

What makes the results obtained with these assumptions still useful are the analyses developed in [Dan70, Dav70, KD75, Maz79]. These works showed that when the step-size is sufficiently small ($\mu \approx 0$), the results obtained using the independence assumptions are good approximations for the actual performance of the LMS algorithm. For example, assume that all data sequences are stationary, and that I-4 holds, but I-1–I-3 do not. Then the *true* steady-state MSE of LMS can be shown to be given by [Maz79]

$$\lim_{k\to\infty} \mathrm{E}\, e(k)^2 = \sigma_v^2 + c_1\mu + c_2\mu^2 + \ldots,$$

where c_1 depends on R. The expression that we obtain from the independence assumptions is [FW85]

$$\lim_{k\to\infty} \mathrm{E}\, e(k)^2 = \sigma_v^2 + c_1\mu + O(\mu^2),$$

where c_1 is as before, and $O(\mu^2)$ denotes a function $f(\mu)$ that decays to zero at least as fast as μ^2, i.e.,

$$\lim_{\mu\to 0} \frac{|f(\mu)|}{\mu^2} < \infty.$$

Thus, we see that the MSE obtained from the independence assumptions is correct up to $O(\mu)$ (i.e., first order in μ — this result holds only when I-4 does).

This conclusion was later extended to more general settings in [Rao81, JCR82]

and [ME83]. Similar results were also obtained for different settings using averaging theory [SK95] and by the ODE method [RM51, Lju77, KY97].

We shall discuss the independence assumptions in much greater details in Chapter 2.

1.4 CONTRIBUTIONS OF THIS WORK

The performance of an adaptive filter is crucially dependent on its rate of convergence, steady-state mean-square error, and stability properties, especially in finite-precision implementations. Exact performance analyses only exist for infinitesimally small step-sizes or under certain so-called independence assumptions. There are practically no counterparts of these analyses for larger step-sizes in the literature. Such results are desirable since they would serve as a guide for the design, and also for a better understanding, of adaptive filters with faster convergence speeds. Progress in this direction is often hindered by the complexity of the (possibly time-variant and nonlinear) update relations that arise when slow adaptation approximations are not employed.

This dissertation develops techniques for the stability and performance analyses of adaptive filters without resorting to slow adaptation approximations. The work expands the four main methods of analysis that have been used so far in the literature, namely, mean-square stability analysis, almost-sure stability analysis, Lyapunov stability analysis, and analysis by simulation or experimentation.

Among the original contributions of this work are the first computable lower bound on the largest step-size that guarantees mean-square stability in the absence of the independence assumptions (Ch. 4); a detailed study of the behavior and properties of ensemble-average learning curves and how special care is needed

in using them to predict or evaluate the performance of an adaptive filter (Ch. 5); a proof that an adaptive filter can actually have two rates of convergence; one rate for the initial phase of operation and another faster rate for later time instants (Ch. 5); a new leakage-based algorithm that avoids both the drift and bias problems of existing adaptive methods (Ch. 6); and a Lyapunov stability analysis for floating-point implementations in worst-case scenarios (Ch. 6). Further contributions to the independence theory itself are provided in Chs. 2 and 3, especially an analysis of the normalized LMS algorithm. More specifically, the following are the contributions of this dissertation.

A) **Independence analysis for singular input covariance matrices.** Usually, the study of the behavior of the LMS algorithm assumes that the regressor covariance matrix, R, is invertible. Nevertheless, in some practical situations (e.g., in adaptive equalization), it can happen that R is singular (or very ill-conditioned). The performance of the LMS algorithm when R is singular is studied in Chapter 2. It is shown that, for any input distribution (with bounded fourth-order moments), the MSE will converge to a small quantity, while the error covariance $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$ in general may be large. This result allows us to explain the possibility of divergence when the algorithm is implemented in finite-precision arithmetic, and in Chapter 6 we shall propose a new algorithm that avoids these problems.

B) **Performance analysis of NLMS.** We give a precise analysis of the convergence and steady-state performance of the NLMS algorithm. The idea is to show how via a suitable change of variables, we can reduce the study of NLMS to that of LMS. This is made possible by noticing that our change of variables leads to a new noise sequence that is still uncorrelated with the new regressor sequence. Although the change of variables itself is already

known, there was no precise analysis of its advantages and consequences.

The results in A) and B) rely on the independence assumptions. While such analyses provide good approximations for the performance of adaptive algorithms for sufficiently small step-sizes ($\mu \approx 0$), it is often desirable in practice to use larger step-sizes in order to improve the convergence speed of an adaptive algorithm.

For larger step-sizes, there are essentially no results in the literature that predict or confirm the behavior/stability of the LMS algorithm (and its variants). Until recently, simulations were the only available tool to predict the behavior of LMS with large μ, as attested by this quote from [Slo93]:

" ... we want to concentrate on two important characteristics of an adaptive filter: its convergence behavior and the steady-state MSE, which remains after the algorithm has converged. *Exact results for both of these items are very scarce, and actually only exist for the asymptotic case of small step-size* μ. (...) when one wants to maximize the convergence speed of the LMS algorithm, a big step size is needed, and especially when one wants to address the issue of the maximum step size for stable operation of the algorithm, *one needs a theory that is valid beyond an infinitesimally small step-size range. At this time, no exact theory of that nature exists.*"

This quote highlights the major contributions of this dissertation. As is well known, there are four main ways to study and analyze the performance of adaptive algorithms, viz.,

1. Mean-square analysis.

2. Almost-sure analysis.

3. Deterministic analysis.

4. Simulations and/or experimentation.

This dissertation studies and expands each of these methods for use with *faster adaptation*. Since performance issues are more naturally treated by mean-square analysis, we devote most of our attention to this kind of analysis, and to the relation of other methods to mean-square results. More specifically, and apart from points A) and B) above, the main original contributions of this dissertation are the following.

C) **Mean-square stability analysis of LMS for non vanishingly-small step-sizes**. Here we build on an approach originally proposed in [FF86] that leads to a state-space framework. Basically, one finds a dynamic state-space model for the evolution of the error covariance matrix $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$. The states are the elements of the covariance matrix, in addition to several other quantities, and the state equation is of the form

$$\boldsymbol{\Gamma}_{k+1} = \Phi\boldsymbol{\Gamma}_k + \boldsymbol{b},$$

where $\boldsymbol{b}$ is a constant vector, $\boldsymbol{\Gamma}_k$ is the state vector, and Φ is a constant matrix. Unfortunately, the dimension of the state-space model grows exponentially fast with the filter length (for $M = 6$ the matrix has size $28,181 \times 28,181$), so the method is computationally feasible only for very small filter length.

For this reason, the reference [FF86] considered only the case $M = 2$ (i.e., a filter with two taps). The reference [DP95] extended the same method for orders up to $M = 6$. However, since the size of Φ increases exponentially

fast with the filter length, it is computationally and analytically infeasible to study the stability of $\boldsymbol{\Gamma}_{k+1} = \Phi\boldsymbol{\Gamma}_k + \boldsymbol{b}$ by working directly with Φ.

We instead show that Φ is both *sparse* and *structured*. These two properties are then used to derive a lower bound on how large the step-size can be for stable performance (in the mean-square sense) in the absence of the independence assumptions. To our knowledge, *this is the first such bound that is computable*. In particular, this is also the first general bound that can be applied when the input sequence is Gaussian.

D) **Convergence speed and learning curves for large step-sizes**. In situations where there are no theoretical results available, or even to confirm the validity of new theoretical results, one usually approximates the MSE performance of an adaptive algorithm by performing several repeated experiments or simulations, and by averaging the results.

If the step-size is small, it is well-known that the average of a few tens of experiments gives a good match with theoretical results (obtained from independence assumptions). We show that the situation is more delicate (and not as immediate) when the step-sizes are large. It may be necessary to perform many thousands of experiments in order to achieve a meaningful result.

Perhaps the most interesting conclusion from our analysis is the fact that the LMS algorithm has a *late* rate of convergence, for large k, that may be considerably *faster* than the rate of convergence for small k. Several examples of these facts are given, and a theoretical analysis for the case of independent inputs is provided.

In fact, as shown in Chapter 5, these results will provide an interpretation of the differences that occur between mean-square and almost-sure perfor-

mance analyses.

E) **New variant of the leaky LMS algorithm with no bias and reduced computational cost**.

In Chapter 6 we review the causes and conditions under which the drift phenomenon can occur, and how the use of the leaky LMS algorithm can avoid drift. Although leaky LMS successfully avoids drift, the price paid is that the estimates become biased and the computational complexity is higher, as compared to that of LMS.

We propose a new algorithm that prevents drift, and yet computes unbiased estimates at essentially the same computational complexity as LMS itself. We call this new algorithm *circular-leaky*, and provide both stability and performance analyses for fixed-point implementations of the algorithm. Our stability analysis is deterministic and borrows from Lyapunov stability theory.

The performance analysis, on the other hand, is stochastic, and is valid only for small step-sizes. By proposing and analyzing this new algorithm, we sidestep an important open problem, viz., the search for a precise analysis of the leaky LMS algorithm. The difficulty of this last task is attested by the following quote from [Set93]:

"Although leakage is one of the most used variants of LMS, it is surprisingly difficult to analyze its behavior precisely. (...) The problem arises because the equilibrium of the system is not independent of the inputs. (...) Because of this dependence of the equilibrium point on the input, it is difficult to carry out the linearization in either the deterministic or stochastic approaches. It remains an open issue how to deal with this situation."

This problem is rendered meaningless, since our new algorithm has the advantages of leaky LMS (without the disadvantages), and a precise analysis of circular-leaky is provided in Chapter 6.

CHAPTER 2

THE INDEPENDENCE ASSUMPTIONS

This chapter studies the behavior of the LMS, NLMS, and leaky LMS algorithms under the independence assumptions. As was mentioned before, the independence assumptions M-1 and I-1–I-4 are seldom satisfied in practice. Nevertheless, it will be shown in Chapter 3 that the results obtained assuming the independence of the inputs (such as rates of convergence, steady-state MSE) are good approximations for the performance of the algorithms in a slow adaptation regime (i.e., small μ).

The independence assumptions M-1 and I-1–I-4 were applied to the study of the LMS algorithm since the 1960s [WMG67, WJM75], for the case of Gaussian variables. The motivation for their use was mainly to obtain a tractable mathematical framework. A more precise treatment, still for the Gaussian case, was later published in [FW85]; and a more elegant solution appeared in [FB88]. An analysis for non-Gaussian variables was presented in [Hsi83], but the results are not as detailed as for Gaussian variables.

The analysis presented here for the LMS algorithm follows the treatments in [FB88] and [SK95] for the Gaussian case. The proofs in the extensions for non-Gaussian variables are similar to [Hsi83] (although the results for small step-sizes are original, they follow from more general theorems in the literature, described in Chapter 3).

The following are the *new* results developed in this chapter.

- A performance analysis of the LMS algorithm when the input covariance matrix $R = \mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T$ is singular (Sec. 2.2). The analysis presented here expands results in the literature, by giving an expression for the MSD (as opposed to only the MSE), and by reducing the number of assumptions made. The case of singular R is important in some applications, as in adaptive equalization.

- A precise analysis of the normalized-LMS algorithm, using a simple change of variables. The analysis of NLMS is more involved than that of LMS, due to the nonlinearity caused by the division by $\|\boldsymbol{x}_k\|^2$. In Sec. 2.4, it is shown that, if a simple change of variables is performed, the NLMS algorithm reduces to the LMS algorithm, but Assumption I-3 is no longer true. The new analysis also shows how to circumvent this difficulty.

- A performance analysis of the leaky LMS algorithm with non-Gaussian input sequences $\{y(k), \boldsymbol{x}_k\}$ (Sec. 2.3.2).

2.1 INDEPENDENCE THEORY FOR LMS

In this section we study the performance of the LMS algorithm, assuming the independence conditions, for the following cases:

i) Stationary and Gaussian inputs.

ii) Stationary and non-Gaussian inputs.

iii) Singular covariance matrices.

The results of iii) are new.

2.1.1 Stationary Gaussian Inputs

In this section, we review the well-known case where the variables $\boldsymbol{x}_k$ and $y(k)$ are assumed to be Gaussian and stationary. Under these conditions, the analysis is simplified by both the stationarity and the use of the following property of Gaussian random variables, viz., if $\boldsymbol{a}$ and $\boldsymbol{b}$ are Gaussian and uncorrelated, they are also independent [Pap84, p. 151].

This property, the orthogonality principle (1.1), and assumptions I-1–I-2 imply I-3 and I-4 of Sec.1.3. Therefore, the following are the conditions we assume here:

M-1. *The sequences* $\{y(k), \boldsymbol{x}_k\}$ *are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2 *and uncorrelated with* $\boldsymbol{x}_k$*.*

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

R-1. *The matrix* R *is positive-definite* ($R > 0$)*.*

G-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly Gaussian.*

IS-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

With these assumptions, the behavior of the LMS algorithm can be described by studying the mean and the covariance of the weight error vector $\tilde{\boldsymbol{w}}_k$ in (1.7). From the LMS recursion (1.6) and recalling that $\tilde{\boldsymbol{w}}_k = \boldsymbol{w}_* - \boldsymbol{w}_k$, the error equation

for the LMS is

$$\tilde{\boldsymbol{w}}_{k+1} = \big(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k - \mu \boldsymbol{x}_k v(k). \tag{2.1}$$

The convergence of $\tilde{\boldsymbol{w}}_k$ in the mean is a simple result, as the following theorem asserts.

Theorem 2.1 (Convergence in the mean). *Under assumptions R-1, I-1–I-2, G-1 and IS-1, the expected value of $\boldsymbol{w}_k$ converges exponentially to $\boldsymbol{w}_*$ if, and only if, the step-size μ satisfies*

$$\mu < \frac{2}{\lambda_{\max}(R)},$$

where $\lambda_{\max}$ denotes the maximum eigenvalue of R.

Proof: Taking expected values of (2.1), we obtain

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1} = \mathrm{E}\,\tilde{\boldsymbol{w}}_k - \mu\,\mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T \tilde{\boldsymbol{w}}_k - \mu\,\mathrm{E}\,\boldsymbol{x}_k v(k).$$

From assumptions M-1, I-1 and I-2, it follows that $\mathrm{E}\,\boldsymbol{x}_k v(k) = \boldsymbol{0}$ and that

$$\mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T \tilde{\boldsymbol{w}}_k = \big(\mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T\big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k.$$

The recursion for the expected value of the weight error becomes

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1} = \Big(I - \mu\,\mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T\Big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k = \big(I - \mu R\big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k.$$

Note that $R = \mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T$ is symmetric and can be diagonalized by an orthogonal transformation, say

$$Q^T R Q = \Lambda = \mathrm{diag}(\lambda_i), \qquad\qquad QQ^T = I. \tag{2.2}$$

Substituting into the recursion for $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$, we obtain

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1} = Q\Big(I - \mu\Lambda\Big)Q^T\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k.$$

From this recursion, it can be seen that $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$ will converge exponentially to zero if and only if $\mu\lambda_{\max}(R) < 2$.

$\diamondsuit$

The fact that the mean weight error $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$ converges to zero does not imply that the algorithm performs well. For good performance, the variance, $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2 - \left(\mathrm{E}\,\|\tilde{\boldsymbol{w}}\|\right)^2$, must also be small. The next (also known) result determines the covariance $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$, and describes the conditions under which this covariance remains bounded. It should be noticed that these conditions are much more restrictive than the conditions in the previous theorem.

Theorem 2.2 (Convergence in the mean-square sense). *Define the covariance matrix*

$$\bar{C}_k = \mathrm{E}\big[\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big]. \tag{2.3}$$

Under assumptions R-1, I-1–I-2, G-1 and IS-1, $\bar{C}_k$ converges to a constant if, and only if,

$$\mu\lambda_i < 1 \qquad \textit{and} \qquad 0 < c \triangleq \frac{1}{2}\sum_{i=1}^{M}\frac{\mu\lambda_i}{1-\mu\lambda_i} < 1, \tag{2.4}$$

where $\lambda_i > 0$ are the eigenvalues of R. The steady-state MSE is given by

$$J = \lim_{k\to\infty}\mathrm{E}\,e(k)^2 = \frac{\sigma_v^2}{1-c},$$

where $e(k) = y(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k$, and $\sigma_v^2 = \mathrm{E}\,v(k)^2$.

The trace of the correlation matrix is the MSD, $\mathcal{D} \triangleq \mathrm{Tr}(\bar{C}_k)$. In steady-state, it evaluates to

$$\lim_{k\to\infty}\mathrm{Tr}(\bar{C}_k) = \mu\frac{\sigma_v^2\sum_{i=1}^{M}\frac{1}{1-\mu\lambda_i}}{2\left(1-\frac{1}{2}\sum_{i=1}^{M}\frac{\mu\lambda_i}{1-\mu\lambda_i}\right)}.$$

For $\mu\lambda_i \ll 1$, we get

$$J \approx \sigma_v^2\Big(1 + \frac{1}{2}\mu \operatorname{Tr} R\Big) \tag{2.5}$$

and

$$\lim_{k\to\infty} \operatorname{Tr}(\bar{C}_k) = \lim_{k\to\infty} \mathrm{E}\, \tilde{\boldsymbol{w}}_k^T \tilde{\boldsymbol{w}}_k \approx \frac{1}{2}\mu M \sigma_v^2. \tag{2.6}$$

Proof: The proof is lengthy, but note that the statement of the theorem asserts that conditions (2.4) are both necessary and sufficient. Moreover, the theorem provides explicit expressions for the steady-state MSE and MSD.

Now multiply the error equation (2.1) by its transpose to obtain

$$\begin{aligned}\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T = \big(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T \big(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\big) + \mu v(k) \boldsymbol{x}_k \tilde{\boldsymbol{w}}_k^T \big(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\big) + \\ + \mu\big(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k \boldsymbol{x}_k^T v(k) + \mu^2 v^2(k) \boldsymbol{x}_k \boldsymbol{x}_k^T.\end{aligned} \tag{2.7}$$

Taking expectations, the cross-terms in $v(k)$ vanish, because of assumptions M-1 and G-1.

Using the independence assumptions, the expected values of almost all quantities are easily found, in terms of R, σ_v^2, and $\bar{C}_k$. The only term that is not readily obtained is

$$\mathrm{E}\Big(\boldsymbol{x}_k \boldsymbol{x}_k^T \big(\boldsymbol{x}_k^T \tilde{\boldsymbol{w}}_k\big)^2\Big), \tag{2.8}$$

which depends on the fourth-order moments of $\boldsymbol{x}_k$. The following property of Gaussian variables is proved in [SK95, p. 295] (see also Appendix 2.A.1). If $\boldsymbol{\alpha}$ is a Gaussian random variable with covariance A, and $\boldsymbol{\beta}$ is a constant vector, then

$$\mathrm{E}\big(\boldsymbol{\alpha}\boldsymbol{\alpha}^T (\boldsymbol{\alpha}^T \boldsymbol{\beta})^2\big) = A\boldsymbol{\beta}^T A \boldsymbol{\beta} + 2A\boldsymbol{\beta}\boldsymbol{\beta}^T A.$$

Using this result, a recursion for $\bar{C}_k$ is readily obtained as

$$\bar{C}_{k+1} = \bar{C}_k - \mu\big(\bar{C}_k R + R\bar{C}_k\big) + 2\mu^2 R\bar{C}_k R + \mu^2 R\,\mathrm{Tr}\big(\bar{C}_k R\big) + \mu^2\sigma_v^2 R. \qquad (2.9)$$

This recursion can be simplified using a change of variables to diagonalize R. With Q still defined as in (2.2), let

$$C_k \triangleq Q^T \bar{C}_k Q. \qquad (2.10)$$

Then

$$C_{k+1} = C_k - \mu\big(C_k\Lambda + \Lambda C_k\big) + 2\mu^2\Lambda C_k\Lambda + \mu^2\,\mathrm{Tr}\big(\Lambda C_k\big)\Lambda + \mu^2\sigma_v^2\Lambda. \qquad (2.11)$$

This is a linear, time-invariant difference equation, though not in standard state-space form. To study its behavior, the diagonal and off-diagonal elements of C_k are considered separately. The recursion for the off-diagonal elements of C_k turns out to be un-coupled, and each element $C_{k,\,ij}$ satisfies

$$C_{k+1,\,ij} = \big(1 - \mu(\lambda_i + \lambda_j) + 2\mu^2\lambda_i\lambda_j\big)C_{k,\,ij}, \quad i \neq j.$$

This means that $C_{k,\,ij} \to 0$ as $k \to \infty$ as long as

$$|1 - \mu(\lambda_i + \lambda_j) + 2\mu^2\lambda_i\lambda_j| < 1. \qquad (2.12)$$

It will be shown shortly that the condition for convergence of the diagonal terms is more restrictive than (2.12).

The recursion for one of the diagonal elements of C_k is

$$\gamma_{k+1,\,i} \triangleq C_{k+1,\,ii} = \big(1 - 2\lambda_i + 2\mu^2\lambda_i^2\big)\gamma_{k,\,i} + \mu^2\lambda_i\Big(\mathrm{Tr}\big(\Lambda C_k\big) + \sigma_v^2\Big).$$

These recursions are coupled. To solve them, collect the diagonal elements of C_k into a vector $\mathbf{\Gamma}_k$,

$$\mathbf{\Gamma}_k \triangleq \begin{bmatrix} C_{k,\,11} & C_{k,\,22} & \ldots & C_{k,\,MM} \end{bmatrix}^T,$$

and define

$$\boldsymbol{L} \triangleq \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_M \end{bmatrix}^T, \qquad \text{and} \qquad A \triangleq 2\mu\Lambda - 2\mu^2\Lambda^2 - \mu^2\boldsymbol{L}\boldsymbol{L}^T.$$

Using these new definitions, the recursion for C_k becomes

$$\boldsymbol{\Gamma}_{k+1} = \big(I - A\big)\boldsymbol{\Gamma}_k + \mu^2\sigma_v^2\boldsymbol{L}. \tag{2.13}$$

For this recursion to reach steady-state we need

$$\big|\lambda_{\max}\big(I - A\big)\big| < 1,$$

where $\lambda(A)$ denotes the set of eigenvalues of A. This condition holds if, and only if, $0 < \lambda\big(A\big) < 2$.

The upper bound on $\lambda(A) < 2$ is always satisfied. Indeed, $\lambda_{\max}(A)$ is

$$\lambda_{\max}\big(2\mu\Lambda - 2\mu^2\Lambda^2 - \mu^2\boldsymbol{L}\boldsymbol{L}^T\big) \leq \lambda_{\max}\big(2\mu\Lambda - 2\mu^2\Lambda^2\big) = 2\mu\lambda_j(R)\big(1 - \mu\lambda_j(R)\big),$$

where j lies in the interval $1 \leq j \leq M$, and the fact that $\boldsymbol{L}\boldsymbol{L}^T$ is positive semi-definite was used.

Denoting $\lambda_j(R)$ by x, the condition $\lambda(A) < 2$ simplifies to

$$2x(1 - x) < 2 \quad \text{for all } x > 0.$$

Since the function $2x(1 - x)$ attains a maximum of $1/2$ for $x = 2$, the inequality above is always satisfied, and it follows that $\lambda(A)$ is always less than 2.

The lower bound ($\lambda(A) > 0$) is satisfied if and only if $A > 0$. Define

$$D = 2\mu\Lambda - 2\mu^2\Lambda^2, \qquad \text{and} \qquad \bar{\boldsymbol{L}} = D^{-1}\boldsymbol{L}.$$

In the following, $X^{(i)}$ denotes the matrix that is obtained by deleting the last $M - i$ rows and columns of an $M \times M$ matrix X, and $\boldsymbol{L}^{(i)}$ denotes the $i \times 1$-vector that is obtained from $\boldsymbol{L}$ by keeping only its first i elements. For example,

if $X = (x)_{i,j} \in \mathbb{R}^{3\times 3}$, $X^{(1)}$ is

$$X^{(1)} = \begin{bmatrix} (x)_{1,1} & (x)_{1,2} \\ (x)_{2,1} & (x)_{2,2} \end{bmatrix}.$$

With this notation, $A > 0$ is equivalent to $\det\big(A^{(i)}\big) > 0$ for $1 \le i \le M$. Therefore, we need

$$\begin{aligned} 0 < \det\big(A^{(i)}\big) &= \det\Big(2\mu\Lambda^{(i)} - 2\mu^2{\Lambda^{(i)}}^2 - \mu^2 \boldsymbol{L}^{(i)}\boldsymbol{L}^{(i),T}\Big) = \\ &= \det\Big(D^{(i)}\Big)\det\Big(I - \mu^2 \bar{\boldsymbol{L}}^{(i)}\boldsymbol{L}^{(i)\,T}\Big) \quad \text{for } 1 \le i \le M. \end{aligned} \tag{2.14}$$

We use a determinant property [Gan60] to evaluate the above expression. For any two column vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, it holds that

$$\det(I - \boldsymbol{a}\boldsymbol{b}^T) = 1 - \boldsymbol{b}^T\boldsymbol{a}.$$

Using this relation, the inequality (2.14) is equivalent to requiring that

$$\det\big(A^{(i)}\big) = \big(2\mu\big)^i \left(1 - \frac{1}{2}\sum_{j=1}^{i} \frac{\mu\lambda_j}{1-\mu\lambda_j}\right) \prod_{j=1}^{i} \lambda_j\big(1-\mu\lambda_j\big) > 0, \quad 1 \le i \le M. \tag{2.15}$$

This condition holds if, and only if,

$$\begin{cases} \mu\lambda_i < 1, \quad \text{for } 1 \le i \le M, \quad \text{and} \\ \dfrac{1}{2}\displaystyle\sum_{i=1}^{M} \frac{\mu\lambda_i}{1-\mu\lambda_i} < 1. \end{cases} \tag{2.16}$$

It is clear that (2.16) implies (2.15). To prove that (2.15) implies (2.16), we proceed by induction. In order that $\det\big(A^{(1)}\big)$ be positive, we must have either

$$1 - \frac{\mu\lambda_1}{2-\mu\lambda_1} > 0 \qquad \text{and} \qquad (2-\mu\lambda_1) > 0,$$

or

$$1 - \frac{\mu\lambda_1}{2-\mu\lambda_1} < 0 \qquad \text{and} \qquad (2-\mu\lambda_1) < 0.$$

This second option cannot occur because if $2-\mu\lambda_1$ is negative, we must necessarily have

$$1 - \frac{\mu\lambda_1}{2-\mu\lambda_1} = 1 + \frac{\mu\lambda_1}{\mu\lambda_1 - 2} > 0.$$

Therefore we must have $\mu\lambda_1 > 0$ and $\frac{\mu\lambda_1}{2-\mu\lambda_1} < 1$. We now assume $\det\left(A^{(2)}\right) > 0$ and conclude that we must have

$$1 - \frac{\mu\lambda_2}{2-\mu\lambda_2} > 0 \qquad \text{and} \qquad (2-\mu\lambda_2) > 0.$$

Proceeding with this argument for the other values of i, we conclude that (2.16) must hold.

We now argue that the condition for convergence of the off-diagonal elements of C_k, (2.12), is satisfied if (2.16) holds. Indeed, letting $\mu\lambda_i = x$ and $\mu\lambda_j = y$, (2.12) reduces to

$$f(x,\, y) = x + y - 2xy < 1.$$

Define the set $\mathcal{A} = \left\{(x,\, y) : \quad 0 \leq x,\, y \leq 1\right\}$, and note that $x,\, y \in \mathcal{A}$ if the eigenvalues $\left\{\mu\lambda_n\right\}_{n=1}^{M}$ satisfy (2.16). It can be shown that

$$\max_{(x,\, y)\in\mathcal{A}} f(x,\, y) = 1,$$

and the maximum is attained only at the boundary of $\mathcal{A}$. This implies that $f(x,\, y) < 1$ inside $\mathcal{A}$, and therefore (2.12) is satisfied.

The steady-state value of $\boldsymbol{\Gamma}_k$ can be evaluated as follows: assuming that (2.16) holds, substitute $\boldsymbol{\Gamma}_\infty$ for $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\Gamma}_{k+1}$ in (2.13) to find the steady-state value

$$\boldsymbol{\Gamma}_\infty = \mu^2\sigma_v^2 A^{-1}\boldsymbol{L}.$$

Using the matrix inversion lemma (see Appendix 2.A.2) to compute the inverse of A, we find

$$A^{-1}\boldsymbol{L} = \big(I - \mu^2 \bar{\boldsymbol{L}}\boldsymbol{L}^T\big)^{-1} D^{-1}\boldsymbol{L} = \big(I - \mu^2 \bar{\boldsymbol{L}}(-1 + \mu^2 \boldsymbol{L}^T \bar{\boldsymbol{L}})^{-1}\boldsymbol{L}^T\big) D^{-1}\boldsymbol{L} =$$
$$= \left(1 + \mu^2 \frac{\boldsymbol{L}^T D^{-1}\boldsymbol{L}}{1 - \mu^2 \boldsymbol{L}^T D^{-1}\boldsymbol{L}}\right) D^{-1}\boldsymbol{L} = \frac{D^{-1}\boldsymbol{L}}{1 - \mu^2 \boldsymbol{L}^T D^{-1}\boldsymbol{L}}.$$

Using this result, the diagonal elements of C_k for $k \to \infty$ can be computed. Recalling that the off-diagonal elements converge to zero, C_∞ is

$$C_\infty = \mu\sigma_v^2 \frac{1}{1 - \frac{1}{2}\sum_{i=1}^{M} \frac{\mu\lambda_i}{1-\mu\lambda_i}} \begin{bmatrix} \frac{1}{2(1-\mu\lambda_1}) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{2(1-\mu\lambda_M)} \end{bmatrix} = \frac{\mu\sigma_v^2}{2(1-c)} \big(I - \mu\Lambda\big)^{-1}.$$

Therefore, the covariance of $\tilde{\boldsymbol{w}}_\infty$ is

$$\mathrm{E}\, \tilde{\boldsymbol{w}}_\infty \tilde{\boldsymbol{w}}_\infty^T = \bar{C}_\infty = \frac{\mu\sigma_v^2}{2(1-c)} \big(I - \mu R\big)^{-1}.$$

Finally, we can compute the MSE as follows,

$$\mathrm{E}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k\big)^2 = \sigma_v^2 + \mathrm{E}\big(\tilde{\boldsymbol{w}}_k^T \boldsymbol{x}_k\big)^2 =$$
$$= \sigma_v^2 + \mathrm{E}\big(\tilde{\boldsymbol{w}}_k^T R \tilde{\boldsymbol{w}}_k\big) = \sigma_v^2 + \mathrm{Tr}\big(R\bar{C}_k\big) =$$
$$= \sigma_v^2 + \mathrm{Tr}\big(\Lambda C_k\big), \tag{2.17}$$

which converges to

$$J = \frac{\sigma_v^2}{1-c}. \tag{2.18}$$

♢

The misadjustment of the algorithm, M_d, measures the relative increase in the steady-state MSE compared to the optimum value σ_v^2, i.e.,

$$M_d = \frac{J - \sigma_v^2}{\sigma_v^2} = \frac{c}{1-c}. \tag{2.19}$$

For small values of μ, this is approximately

$$M_d \approx \frac{\mu}{2} \operatorname{Tr} R. \tag{2.20}$$

2.1.2 Stationary Non-Gaussian Inputs

As mentioned earlier, the performance analysis in this section is similar to a result in [Hsi83]. It is based on relaxing the requirement G-1 of Gaussian variables. But we need to re-incorporate assumptions I-3–I-4, as well as assume the following.

IS-2. *The fourth-order moments of $\boldsymbol{x}_k$ are bounded by a constant B, i.e.,*

$$\mathrm{E}\big(\boldsymbol{x}_k^T \boldsymbol{x}_k\big)^2 < B < \infty.$$

In summary, in this section we assume that IS-2 holds, as well as:

M-1. *The sequences $\big\{y(k),\, \boldsymbol{x}_k\big\}$ are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$, and where $v(k)$ is zero-mean with variance σ_v^2.

I-1. *The sequence $\big\{\boldsymbol{x}_k\big\}$ is independent.*

I-2. *$y(k)$ is correlated with $\boldsymbol{x}_k$, but is independent of all $\boldsymbol{x}_j$ with $j \neq k$.*

I-3. *The noise sequence $\big\{v(k)\big\}$ is independent of the input sequence.*

I-4. *The noise sequence $\big\{v(k)\big\}$ is independent and identically distributed (iid).*

R-1. *The matrix R is positive-definite ($R > 0$).*

IS-1. *The random sequences $\{\boldsymbol{x}_k\}$ and $\{y(k)\}$ are jointly stationary.*

Theorem 2.1 still holds, but Theorem 2.2 has to be modified. If the only information we have available about the sequence $\{\boldsymbol{x}_k\}$ is the correlation matrix R, then the equivalent of Thm. 2.2 will be weaker in the sense that the result will only hold for $\mu \approx 0$. Stronger results can be obtained if the input distribution is known, as we show in Theorem 2.4 further ahead.

The idea is to simplify the computation of $\mathrm{E}\left(\boldsymbol{x}_k \boldsymbol{x}_k^T \bar{C}_k \boldsymbol{x}_k \boldsymbol{x}_k^T\right)$ in (2.8) using *Kronecker products*. The Kronecker product of two matrices $A \in \mathbb{R}^{m_a \times n_a}$ and $B \in \mathbb{R}^{m_b \times n_b}$ is defined as [HJ94]

$$A \otimes B = \begin{bmatrix} a_{1,1}B & a_{1,2}B & \dots & a_{1,n_a}B \\ a_{2,1}B & a_{2,2}B & \dots & a_{2,n_a}B \\ \vdots & \vdots & & \vdots \\ a_{m_a,1}B & a_{m_a,2}B & \dots & a_{m_a,n_a}B \end{bmatrix}. \tag{2.21}$$

This operation has several useful properties, but the one that interests us most is the following. Define the symbol $\mathrm{vec}(A)$ to represent an $m_a n_a$ column vector formed by stacking the columns of the matrix A one above the other. Let $C = AXB$, where A, B, and X are matrices of compatible dimensions. Then the following equality holds [HJ94, p. 254]

$$\mathrm{vec}(C) = \left(B^T \otimes A\right) \mathrm{vec}(X). \tag{2.22}$$

Using these definitions, the next two theorems can be proved. Although the proof of Theorem 2.3 below is new, the theorem follows directly from more general results from the literature (as shown in Sec. 3.2). Theorem 2.4 is an extension allowing for larger values of μ.

Theorem 2.3 (Steady-state covariance). *The covariance matrix*

$$\bar{C}_k = \mathrm{E}\big[\tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T\big]$$

converges to a constant if Assumptions R-1, I-1–I-4, IS-1 and IS-2 hold, and if the step-size satisfies $\mu B \ll \lambda_{\min}(R)$. *If* $\mu \approx 0$, *the following approximations hold*

$$J \approx \sigma_v^2\Big(1 + \frac{1}{2}\mu \operatorname{Tr} R\Big), \tag{2.23}$$

$$\mathcal{D} = \lim_{k\to\infty} \operatorname{Tr}(\bar{C}_k) \approx \frac{1}{2}\mu M \sigma_v^2. \tag{2.24}$$

In addition, the rate of convergence of the covariance matrix $\bar{C}_k$ *is approximately determined by the eigenvalues of* $(I - 2\mu R)$.

Proof: The main point of this proof is to use Assumption IS-2 to ignore the third term in the recursion for $\bar{C}_k$ below,

$$\bar{C}_{k+1} = \bar{C}_k - \mu R\bar{C}_k - \mu \bar{C}_k R + \mu^2\bigg(\mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T \bar{C}_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big)\bigg) + \mu^2 \sigma_v R. \tag{2.25}$$

Recall that in order to obtain the recursion (2.9) for $\bar{C}_k$, we needed to evaluate the term

$$F(\bar{C}_k) \stackrel{\Delta}{=} \mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T \bar{C}_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big).$$

The computation of $F(\bar{C}_k)$ was the only place where Assumption G-1 was used in the proof of Theorem 2.2. Unless the distribution of $\boldsymbol{x}_k$ is known, it is impossible to evaluate F explicitly. However, we can analyze the recursion for $\bar{C}_k$ for small μ, using Kronecker products and without knowing the distribution of $\boldsymbol{x}_k$.

Indeed, applying (2.22) to (2.25) we obtain the recursion

$$\begin{aligned}\operatorname{vec}\big(\bar{C}_{k+1}\big) = \Big(I_{M^2} - \mu R \otimes I_M - \mu I_M \otimes R\Big) \operatorname{vec}\big(\bar{C}_k\big) + \\ + \mu^2 \operatorname{vec}\big(F\big(\bar{C}_k\big)\big) + \mu^2 \sigma_v^2 \operatorname{vec}\big(R\big),\end{aligned} \tag{2.26}$$

where I_r represents the identity matrix of dimensions $r \times r$.

Note that $\mathrm{vec}\left(F\left(\bar{C}_k\right)\right)$ is a linear function of $\mathrm{vec}\left(\bar{C}_k\right)$, since

$$\mathrm{vec}\left(F\left(\bar{C}_k\right)\right) = \mathrm{E}\left[\left(\boldsymbol{x}_k\boldsymbol{x}_k^T \otimes \boldsymbol{x}_k\boldsymbol{x}_k^T\right)\mathrm{vec}\left(\bar{C}_k\right)\right] \triangleq G\,\mathrm{vec}\left(\bar{C}_k\right).$$

From its definition, G is a symmetric and nonnegative-definite matrix.

To study the stability of (2.26), we use another property of Kronecker products, viz., if $\left\{\nu_i\right\}_{i=1}^n$ and $\left\{\rho_j\right\}_{j=1}^m$ are the eigenvalues of, respectively, the square matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{m\times m}$, then the eigenvalues of $\left(I_m \otimes A\right) + \left(B \otimes I_n\right)$ are the pairs $\nu_i + \rho_j$, for all (i,j) with $1 \le i \le n$ and $1 \le j \le m$ [HJ94, p. 268].

This implies that the eigenvalues of $\left(I_{M^2} - \mu R \otimes I_M - \mu I_M \otimes R\right)$ are equal to $1 - \mu\lambda_m - \mu\lambda_n$, where λ_p are the eigenvalues of R and $1 \le m,\, n,\, p \le M$.

To proceed, we need a bound for the norm of G. Let $\boldsymbol{e}_i$ be the i-th canonical vector of length M, and Y be a matrix with the same dimensions as $\bar{C}_k$. The (i,j)-th element of $F\left(\mathrm{vec}(Y)\right)$ is bounded by

$$\left|\mathrm{E}\left(\boldsymbol{e}_i^T\boldsymbol{x}_k\boldsymbol{x}_k^T Y\,\boldsymbol{x}_k\boldsymbol{x}_k^T\boldsymbol{e}_i\right)\right| \le \mathrm{E}\left(\|\boldsymbol{x}_k\|^4\|Y\|\right) \le B\|Y\|,$$

where $\|\cdot\|$ denotes the Euclidean norm of a vector or the maximum singular value of a matrix. Using the property of norms $\|A\|_F \le M\|A\|$ for any matrix A, it follows from the above relation that

$$\begin{aligned}\left\|G\,\mathrm{vec}\left(Y\right)\right\| = \left\|\mathrm{vec}\left(F\left(Y\right)\right)\right\| = \left\|F\left(Y\right)\right\|_F \le \\ \le MB\left\|Y\right\| \le MB\left\|Y\right\|_F \triangleq \beta\left\|\mathrm{vec}\left(Y\right)\right\|,\end{aligned}$$

where $\|\cdot\|_F$ denotes the Frobenius norm (this is a very loose bound, but it is enough for the purposes of this theorem).

The stability of (2.26) is determined by the eigenvalues of the matrix

$$A \triangleq I_{M^2} - \mu R \otimes I_M - \mu I_M \otimes R + \mu^2 G.$$

Since A is symmetric, its largest eigenvalue is given by

$$\max_{\|\boldsymbol{y}\|=1} \boldsymbol{y}^T A \boldsymbol{y} \leq \max_{\|\boldsymbol{y}\|=1} \boldsymbol{y}^T \Big(I_{M^2} - \mu R \otimes I_M - \mu I_M \otimes R \Big) \boldsymbol{y} + \mu^2 \beta =$$
$$= 1 - 2\mu\lambda_{\min} + \mu^2 \beta,$$

where $\lambda_{\min}$ is the smallest eigenvalue of R. We conclude that A is stable if R is positive-definite ($\lambda_M > 0$) and μ is sufficiently small. In addition, if A is stable and μ is small enough, the steady-state $\mathrm{vec}\big(\bar{C}_\infty\big)$ satisfies

$$\Big(\mu R \otimes I_M + \mu I_M \otimes R \Big) \mathrm{vec}\big(\bar{C}_\infty\big) \approx \Big(\mu R \otimes I_M + \mu I_M \otimes R - \mu^2 G \Big) \mathrm{vec}\big(\bar{C}_\infty\big) =$$
$$= \mu^2 \sigma_v^2 \, \mathrm{vec}\big(R\big). \tag{2.27}$$

Since for small μ and $R > 0$ the matrix $\Big(R \otimes I_M + I_M \otimes R - \mu G \Big)$ is non-singular, we conclude that

$$\mathrm{vec}\big(\bar{C}_\infty\big) = \mu^2 \sigma_v^2 \big(I_{M^2} - A \big)^{-1} \mathrm{vec}\big(R\big) \approx \mu \sigma_v^2 \Big(R \otimes I_M + I_M \otimes R \Big)^{-1} \mathrm{vec}\big(R\big).$$

This inverse can be computed as follows. Applying (2.22) to (2.27), we obtain the relation

$$R\bar{C}_\infty + \bar{C}_\infty R \approx \mu \sigma_v^2 R.$$

Multiplying this relation on the left by Q^T and on the right by Q, where $Q^T R Q = \Lambda$ and Λ is diagonal (2.2), we obtain

$$\Lambda C_\infty + C_\infty \Lambda \approx \mu \sigma_v^2 \Lambda,$$

where, as before, $C_\infty = Q^T \bar{C}_\infty Q$.

From the above relation, we conclude that C_∞ is approximately diagonal, with the diagonal entries approximately given by $(C_\infty)_{i,i} \approx \mu\sigma_v^2/2$, i.e.,

$$C_\infty \approx \frac{\mu \sigma_v^2}{2} I.$$

The MSD and MSE follow directly from this expression.

$\diamondsuit$

If the distribution of $\boldsymbol{x}_k$ is known, we can give a more detailed analysis. The proof for the theorem below was actually already given in the proof of Thm. 2.3, but we state this result separately since it provides a solution for the performance of the LMS algorithm with general distributions of the (iid) input sequence $\{\boldsymbol{x}_k\}$, valid for all values of the step-size. Note that the step-size does not need to be small in the statement of the theorem.

Theorem 2.4. *If all fourth-order moments of $\boldsymbol{x}_k$ are known, and under the Assumptions R-1, I-1–I-4, IS-1 and IS-2, the weight-error covariance $\bar{C}_k$ can be computed by the recursion:*

$$\begin{aligned}\mathrm{vec}(\bar{C}_{k+1}) = &\left[I_{M^2} - \mu\big(R \otimes I_M\big) - \mu\big(I_M \otimes R\big) + \mu^2\,\mathrm{E}\left(\boldsymbol{x}_k\boldsymbol{x}_k^T \otimes \boldsymbol{x}_k\boldsymbol{x}_k^T\right)\right]\mathrm{vec}(\bar{C}_k) + \\ &+ \mu^2\sigma_v^2\,\mathrm{vec}(R), \end{aligned} \tag{2.28}$$

where the initial condition is $\bar{C}_0 = \mathrm{E}\,\tilde{\boldsymbol{w}}_0\tilde{\boldsymbol{w}}_0^T$ (which we assume known).

$\diamondsuit$

We do not give expressions for the steady-state MSE and MSD, since these will depend on the particular distribution of $\boldsymbol{x}_k$. Nevertheless, these values can be computed once (2.28) is obtained. As an example, assume that $\{\boldsymbol{x}_k \in \mathbb{R}^2\}$ is a sequence of iid (independent and identically-distributed) random vectors,

$$\boldsymbol{x}_k = \begin{bmatrix} a_1(k) & a_2(k) \end{bmatrix}^T,$$

whose entries $a_i(k)$ are mutually independent and uniformly distributed between

-1 and 1. It follows that $R = 1/3\ I_2$,

$$I_2 \otimes R = R \otimes I_2 = \frac{1}{3} I_4, \qquad \text{and} \qquad \mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T \otimes \boldsymbol{x}_k \boldsymbol{x}_k^T\big) = \begin{bmatrix} \frac{1}{5} & 0 & 0 & \frac{1}{9} \\ 0 & \frac{1}{9} & \frac{1}{9} & 0 \\ 0 & \frac{1}{9} & \frac{1}{9} & 0 \\ \frac{1}{9} & 0 & 0 & \frac{1}{5} \end{bmatrix}.$$

With these values, and taking $\mu = 0.01$, we can compute $\bar{C}_\infty$ by setting $\bar{C}_\infty = \bar{C}_{k+1} = \bar{C}_k$ in (2.28), obtaining

$$\bar{C}_\infty = \sigma_v^2 \begin{bmatrix} 0.005023 & 0 \\ 0 & 0.005023 \end{bmatrix} \approx \frac{\mu \sigma_v^2}{2} I.$$

On the other hand, for $\mu = 0.5$, we have

$$\bar{C}_\infty = \sigma_v^2 \begin{bmatrix} 0.3261 & 0 \\ 0 & 0.3261 \end{bmatrix},$$

which is substantially different than $0.25\sigma_v^2 I$.

2.2 SINGULAR INPUT COVARIANCE MATRIX

When the Assumption R-1 does not hold, or when R is ill-conditioned, it still can be shown that the MSE remains bounded and converges to a small quantity. In fact, even the MSE convergence rate will not appear to be excessively slow if R is very close to singular — but the MSD will be affected in an undesirable way, as we show in (2.31) and (2.37) below.

The case of singular R arises in some applications, especially in channel equalizers for which the sampling rate at the receiver input is higher than the transmitter symbol rate (such equalizers are called "fractionally-spaced equalizers", or FSE's, see [QG77, GMW82, Pro89, GHW92, TFC96]). One example where

the input covariance matrix is singular can be found in [GMW82]. We present a modified version of the example here, showing that an FSE whose input has bandwidth equal to $2/T$ (where T is the symbol rate) may lead to a singular covariance matrix.

Figure 2.2 depicts an equalization scheme with a fractionally-spaced equalizer. The equalizer sampling period, T', is equal to one-half the symbol rate T. The input to the equalizer is the vector

$$\boldsymbol{x}_k = \Big[a(k-M+1) \quad a(k-M+2) \quad \ldots \quad a(k)\Big]^T,$$

where the sequence $\big\{a(k)\big\}$ is obtained by sampling the filtered received signal, $d(t)$, with a rate $1/T' = 2/T$, i.e.,

$$a(k) = d(T'k).$$

Denote by $D(\omega)$ the power spectrum of $d(t)$, i.e.,

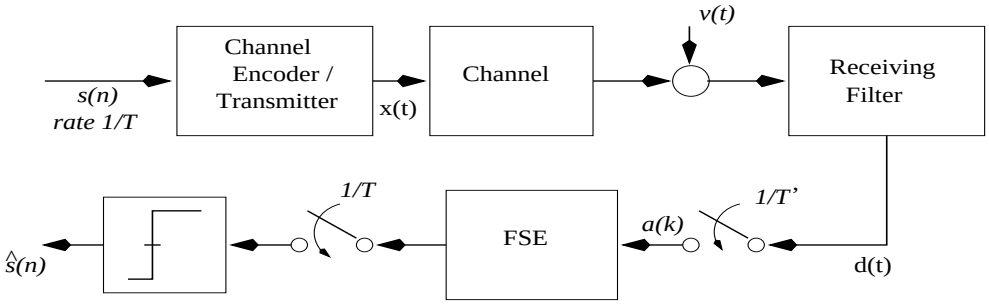

Figure 2.1: *Channel equalization with a fractionally-spaced equalizer.*

$$D(\omega) = \int_{-\infty}^{\infty} R_d(\tau) e^{-j\omega\tau} \, \mathrm{d}\, \tau,$$

where $j = \sqrt{-1}$, and $R_d(\tau)$ is the autocorrelation function of $d(t)$,

$$R_d(\tau) = \mathrm{E}\, d(t) d(t-\tau).$$

We will find conditions under which the covariance matrix $R \stackrel{\Delta}{=} \mathrm{E}\, \boldsymbol{x}_k \boldsymbol{x}_k^T$ is singular. Since R is positive-semidefinite, it is singular if, and only if, there exists a vector $\boldsymbol{u}$ such that $\boldsymbol{u}^T R \boldsymbol{u} = 0$. Let us evaluate this last quantity for a generic $\boldsymbol{u}$:

$$\boldsymbol{u}^T R \boldsymbol{u} = \mathrm{E}\left(\boldsymbol{u}^T \boldsymbol{x}_k \boldsymbol{x}_k^T \boldsymbol{u}\right) = \mathrm{E}\left(\sum_{m=1}^{M}\sum_{n=1}^{M} u_m u_n a(k-m+1)a(k-n+1)\right) =$$
$$= \sum_{m=1}^{M}\sum_{n=1}^{M} u_m u_n R_d\big((m-n)T'\big).$$

Now we use the inverse Fourier transform to write

$$\boldsymbol{u}^T R \boldsymbol{u} = \sum_{m=1}^{M}\sum_{n=1}^{M}\left(u_m u_n \int_{-\infty}^{\infty} D(\omega) e^{j(m-n)T'\omega}\, \frac{\mathrm{d}\,\omega}{2\pi}\right) =$$
$$= \int_{-\infty}^{\infty} D(\omega)\left(\sum_{m=1}^{M} u_m e^{jm\omega T'}\right)\left(\sum_{n=1}^{M} u_n e^{-jn\omega T'}\right)\frac{\mathrm{d}\,\omega}{2\pi}.$$

Define the function

$$U(\omega) \stackrel{\Delta}{=} \sum_{m=1}^{M} u_m e^{-j\, m\omega T'},$$

and note that $U(\omega)$ is periodic with period $1/T'$. We conclude that R is singular if, and only if,

$$\int_{-\infty}^{\infty} |U(\omega)|^2 D(\omega)\frac{\mathrm{d}\,\omega}{2\pi} = 0, \tag{2.29}$$

where $|U(\omega)|^2$ is periodic with period $2/T$.

We now present an example satisfying condition (2.29). Let $D(\omega)$ have bandwidth $2/T$ and satisfy

$$D(\omega) = \begin{cases} 1, & 0 \le |\omega| \le \frac{2\pi}{T}, \\ -1, & \frac{2\pi}{T} < |\omega| < \frac{4\pi}{T}, \\ 0, & \text{otherwise.} \end{cases}$$

Assume that $|U(\omega)|^2$ is symmetric around $\omega = 1/T$. It then follows that condition (2.29) will be satisfied, and thus R will be singular.

$\diamondsuit$

Another situation where singular input covariance matrices appear is in the estimation of multiple sinusoids, where the rank of the correlation matrix is equal to the number of sinusoids [TK82].

In fact, exactly singular input covariance matrices are rare in practice, but covariance matrices very close to singular are common, as explained in [GMW82]. Even an ill-conditioned, but non-singular covariance matrix may lead to overflow (and poor performance) when the filters are implemented in finite-precision arithmetic (this can be seen from the term $(2\mu\Lambda')^{-1}\boldsymbol{L}_\xi$ in (2.B.4) in Appendix 2.B). When the input sequence $\{\boldsymbol{x}_k\}$ is non-stationary, the possibility of filter instability increases. In Chapter 6 we discuss the possibility of instability in LMS with detail.

Our result in this section proves that a singular or ill-conditioned R will not affect the steady-state MSE (a fact that is known from simulations and practice). This fact was proved, under a more restrictive (and unnecessary) set of assumptions in [Lin88] (see the comments after (2.32)). That reference only studied the MSE, while we extend the results to provide an expression for the MSD as well.

The theorem uses the following conditions.

M-1. *The sequences* $\{y(k),\, \boldsymbol{x}_k\}$ *are related through a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2*.*

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-3. *The noise sequence* $\{v(k)\}$ *is independent of the input sequence.*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

R-1. *The matrix* R *is positive-definite* ($R > 0$)*.*

IS-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

IS-2. *The fourth-order moments of* $\boldsymbol{x}_k$ *are bounded by a constant* B*, i.e.,*

$$\mathrm{E}\big(\boldsymbol{x}_k^T \boldsymbol{x}_k\big)^2 < B < \infty.$$

We now have the following.

Theorem 2.5. *Assume that I-1–I-4 and IS-1 hold, and that the eigenvalues of* R *are*

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_K > \lambda_{K+1} = \cdots = \lambda_M = 0.$$

Then for a sufficiently small step-size μ*, the MSE computed by the LMS algorithm is approximately*

$$\lim_{k\to\infty} \mathrm{E}\, e(k)^2 \approx \sigma_v^2 + \mu \frac{\sigma_v^2}{2} \operatorname{Tr}(R), \tag{2.30}$$

and the correlation $\mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T$ *satisfies*

$$\lim_{k\to\infty} \bar{C}_k \approx Q^T \begin{bmatrix} \mu\frac{\sigma_v^2}{2} I & 0 \\ 0 & C_0^{(2,\,2)} \end{bmatrix} Q, \tag{2.31}$$

where $C_0^{(2,\,2)}$ *is the lower* $M-K \times M-K$ *block of*

$$C_0 \triangleq Q^T \,\mathrm{E}\left(\tilde{\boldsymbol{w}}_0 \tilde{\boldsymbol{w}}_0^T\right) Q,$$

and Q is an orthogonal matrix that diagonalizes R — see (2.2). In addition, the steady-state MSD is

$$\mathcal{D} = \mu \frac{K}{2} \sigma_v^2 + \mathrm{Tr}\left(C_0^{(2,2)}\right).$$

Proof: To simplify the analysis, assume without loss of generality that a change of variables has been performed such that (see (2.2))

$$\mathrm{E}\, \boldsymbol{x}_k \boldsymbol{x}_k^T = \Lambda = \mathrm{diag}(\lambda_i). \tag{2.32}$$

In [Lin88], it was assumed that the entries of the rotated $\boldsymbol{x}_k$ above are not only uncorrelated, but also independent. As the analysis below shows, this is an unnecessarily restrictive assumption.

From (2.18), the MSE computed by the LMS algorithm is (note that the relation below still holds if Λ is singular)

$$\mathrm{E}\, e(k)^2 = \sigma_v^2 + \mathrm{Tr}\big(\Lambda C_k\big),$$

where now (using (2.32)), $C_k = \mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T = \bar{C}_k$.

Recall the recursion (2.25) for C_k:

$$C_{k+1} = C_k - \mu C_k \Lambda - \mu \Lambda C_k + \mu^2\, \mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big) + \mu^2 \sigma_v^2 \Lambda,$$

where we replaced R by Λ. The argument used in Theorem 2.3 to ignore the term

$$E\big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big)$$

must be modified, since now Λ is singular. The core of the argument below is that if $\boldsymbol{y}$ is an eigenvector of Λ relative to the eigenvalue 0, then $E\big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big)\boldsymbol{y} = \mathbf{0}$.

Indeed, let $\boldsymbol{e}_i$ be the the i-th canonical vector. From its definition, Λ satisfies

$$\boldsymbol{e}_i^T \Lambda \boldsymbol{e}_i = 0, \quad \text{for } i > K,$$

and thus

$$\mathrm{E}\big(\boldsymbol{e}_i^T \boldsymbol{x}_k\big)^2 = 0, \quad \text{for } i > K. \tag{2.33}$$

Let the probability space over which $\boldsymbol{x}_k$ is defined be (Ω, P, $\mathcal{F}$), where Ω is the set of possible outcomes of the random variable $\boldsymbol{x}_k$, $\mathcal{F}$ is a σ-algebra defined over that set, and P is a probability measure defined on (Ω, $\mathcal{F}$) [Dur96]. Since $(\boldsymbol{e}_i^T \boldsymbol{x}_k)^2$ is nonnegative, (2.33) implies that the events

$$\mathcal{A}_i = \big\{\boldsymbol{x}_k : \quad (\boldsymbol{e}_i^T \boldsymbol{x}_k) \neq 0\big\} \tag{2.34}$$

have probability zero for $i > K$. Therefore, the events

$$\mathcal{B}_i = \big\{\boldsymbol{x}_k : \quad \big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T \boldsymbol{e}_i\big) \neq \mathbf{0}\big\}, \quad i > K$$

have also zero probability. This means that $\boldsymbol{e}_i$ is an eigenvector of

$$\mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big)$$

with an eigenvalue of 0. Moreover, since the matrix is symmetric, it can be decomposed as

$$\mathrm{E}\big(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\big) = \begin{bmatrix} D_k & 0 \\ 0 & 0 \end{bmatrix},$$

where $D_k \in \mathbb{R}^{K \times K}$ is a linear function of C_k.

Now write Λ as

$$\Lambda = \begin{bmatrix} \bar{\Lambda} & 0 \\ 0 & 0 \end{bmatrix},$$

where $\bar{\Lambda} = \mathrm{diag}(\lambda_i)$, $1 \leq i \leq K$. The recursion for C_k then becomes

$$\begin{bmatrix} C_{k+1}^{(1,1)} & C_{k+1}^{(1,2)} \\ C_{k+1}^{(2,1)} & C_{k+1}^{(2,2)} \end{bmatrix} = \begin{bmatrix} C_k^{(1,1)} & C_k^{(1,2)} \\ C_k^{(2,1)} & C_k^{(2,2)} \end{bmatrix} - \mu \begin{bmatrix} \bar{\Lambda} C_k^{(1,1)} & \bar{\Lambda} C_k^{(1,2)} \\ 0 & 0 \end{bmatrix} - \mu \begin{bmatrix} C_k^{(1,1)} \bar{\Lambda} & 0 \\ C_k^{(2,1)} \bar{\Lambda} & 0 \end{bmatrix} + \\ + \mu^2 \begin{bmatrix} D_k & 0 \\ 0 & 0 \end{bmatrix} + \mu^2 \sigma_v^2 \begin{bmatrix} \bar{\Lambda} & 0 \\ 0 & 0 \end{bmatrix},$$

where D_k is a linear function of $C_k^{(1,1)}$, $C_k^{(1,2)}$, $C_k^{(2,1)}$, and $C_k^{(2,2)}$.

Therefore, the recursions for the blocks $C_k^{(1,2)}$ and $C_k^{(2,1)}$ are un-coupled from the others, and converge to zero if the step-size satisfies

$$|1 - \mu \max_i \lambda_i| < 1. \tag{2.35}$$

The recursion for $C_k^{(2,2)}$ is simply

$$C_{k+1}^{(2,2)} = C_k^{(2,2)},$$

i.e., this block remains forever equal to its initial value.

Finally, the (1, 1) block is

$$C_{k+1}^{(1,1)} = C_k^{(1,1)} - \mu \bar{\Lambda} C_k^{(1,1)} - \mu C_k^{(1,1)} \bar{\Lambda} + \mu^2 D_k + \mu^2 \sigma_v^2 \bar{\Lambda}. \tag{2.36}$$

It is possible to conclude directly from this recursion that $C_k^{(1,1)}$ remains bounded. Better results can be obtained if the structure of D_k is further studied. In fact, D_k is the $(1,1)$ block of

$$\mathrm{E}\left(\boldsymbol{x}_k \boldsymbol{x}_k^T C_k \boldsymbol{x}_k \boldsymbol{x}_k^T\right) = \mathrm{E}\left(\boldsymbol{x}_k \boldsymbol{x}_k^T \left(\begin{bmatrix} C_k^{(1,1)} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & C_k^{(1,2)} \\ C_k^{(2,1)} & 0 \end{bmatrix} + \right.\right. \\ \left.\left. + \begin{bmatrix} 0 & 0 \\ 0 & C_k^{(2,2)} \end{bmatrix} \right) \boldsymbol{x}_k \boldsymbol{x}_k^T \right).$$

From (2.34), it follows that

$$\boldsymbol{x}_k \boldsymbol{x}_k^T \begin{bmatrix} 0 & 0 \\ 0 & C_k^{(2,2)} \end{bmatrix} \boldsymbol{x}_k \boldsymbol{x}_k^T = 0 \quad \text{with probability 1.}$$

This implies that D_k is a *linear* function of $C_k^{(1,1)}$, $C_k^{(1,2)}$, and $C_k^{(2,1)}$ *only*. In addition, if condition (2.35) is satisfied, $C_k^{(1,2)}$ and $C_k^{(2,1)}$ decay exponentially fast to zero. Therefore, the recursion (2.36) for the $C_k^{(1,1)}$-block is very similar to (2.25), with $\bar{\Lambda} > 0$, but with an extra term that decays exponentially to zero.

Since (2.36) is linear, these exponentially decaying terms will have no effect on the steady-state solution, and the arguments of the proof of Thm. 2.3 can be repeated to show that for sufficiently small μ, the recursion for $C_k^{(1,1)}$ converges exponentially fast to the steady-state value

$$\lim_{k\to\infty} C_k^{(1,1)} = C_\infty^{(1,1)} \approx \mu \frac{\sigma_v^2}{2} I.$$

$\diamondsuit$

Therefore, even if Λ is not invertible, the performance of the LMS algorithm appears to be quite good. Unfortunately, however, when the algorithm is implemented in finite-precision arithmetic, there is an additional driving term that is not proportional to Λ (see Eq. (2.B.3) in Appendix 2.B). In this case, $C_k^{(1,1)}$ remains bounded, $C_k^{(1,2)}$ and $C_k^{(2,1)}$ still converge to 0, but the recursion for $C_k^{(2,2)}$ becomes

$$C_{k+1}^{(2,2)} = C_k^{(2,2)} + \Lambda_\xi^{(2,2)}, \tag{2.37}$$

where $\Lambda_\xi^{(2,2)}$ depends on the size of the word-length used in the implementation of the filter. Therefore, $C_k^{(2,2)} \to \infty$ as $k \to \infty$.

However, the MSE remains bounded. In fact, it is given by

$$\mathrm{E}\, e(k)^2 = \sigma_v^2 + \mathrm{Tr}(\Lambda C_k),$$

and since Λ is singular, the diverging part of C_k does not influence $\mathrm{E}\, e(k)^2$.

The divergence of $C_k^{(2,2)}$ may cause the registers where $\boldsymbol{w}_k$ is stored eventually overflow, causing a considerable performance degradation. This problem (also known as the *drift problem* of LMS) led to the introduction of the leaky LMS algorithm. The drift problem is discussed in more detail in Chapter 6, and also in [GMW82, IK84, SLJ86, Set92, Set93].

2.3 INDEPENDENCE THEORY FOR LEAKY LMS

The leaky LMS algorithm has been proposed to resolve the drift problem of LMS. It nevertheless introduces problems of its own (such as biased estimates and increased computational cost). In Chapter 6 we shall develop a new algorithm that resolves these difficulties.

Meanwhile, since leaky LMS is a popular algorithm, we provide in this section a performance analysis, assuming the independence conditions, for the following cases:

i) Stationary and Gaussian inputs.

ii) Stationary and non-Gaussian inputs.

The results of ii) are new. The results of i) have been recently derived in [Say97], and are new in that they provide *both* necessary and sufficient conditions for convergence. A weaker analysis, with only sufficient conditions, appeared in [MA97].

2.3.1 Stationary Gaussian Inputs

The analysis for the leaky LMS algorithm is similar to that for LMS, but the arguments are longer. The most important difference is that $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ does not converge to zero, i.e., the algorithm computes a *biased* estimate.

In this section, the following assumptions are made:

M-1. *The sequences* $\{y(k),\, \boldsymbol{x}_k\}$ *are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2*.*

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

G-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly Gaussian.*

IS-3. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

Recall that the leaky LMS recursion is given by (1.9):

$$\boldsymbol{w}_{k+1}^l = (1 - \mu\alpha_0)\boldsymbol{w}_k^l + \mu\boldsymbol{x}_k\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k\big), \quad \text{with initial condition } \boldsymbol{w}_0.$$

The necessary and sufficient conditions for convergence of the error covariance matrix in the following statement were recently proved in [Say97]. A weaker, sufficient condition only, was given in [MA97].

Theorem 2.6. *Under the assumptions I-1, I-2, G-1, and IS-1, the mean* $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ *converges to a steady-state if and only if the step-size satisfies*

$$\mu < \frac{2}{\alpha_0 + \lambda_{\max}(R)}, \tag{2.38}$$

The steady-state error computed by the leaky LMS algorithm is given by

$$\lim_{k\to\infty} \mathrm{E}\,\tilde{\boldsymbol{w}}_k^l = \alpha_0\big(\alpha_0 I + R\big)^{-1}\boldsymbol{w}_*.$$

In addition, let Q be an orthogonal matrix that diagonalizes R, as in (2.2), and let the eigenvalues of R be ordered such that

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_M \geq 0.$$

The covariance $\bar{C}_k$ converges to a finite constant if and only if the step-size satisfies

$$\mu < \frac{2}{\alpha_0 + \lambda_1}, \qquad 0 < c^\alpha = \sum_{j=1}^{M} \frac{\mu\lambda_j^2}{(\alpha_0+\lambda_j)\big[2-\mu(\alpha_0+\lambda_j)\big]} < 1.$$

Under these conditions, the steady-state MSE is

$$\lim_{k\to\infty} \mathrm{E}\,e(k)^2 = \sigma_v^2 + \frac{1}{1-c^\alpha}\left[\sum_{j=1}^{M} \frac{\mu\sigma_v^2\lambda_j^2}{(\alpha_0+\lambda_j)\big[2-\mu(\alpha_0+\lambda_j)\big]} + \sum_{j=1}^{M} \frac{\alpha_0^2\big(\bar{\boldsymbol{w}}_*\big)_j^2}{\lambda_j(\alpha_0+\lambda_j)^2}\right],$$

where $\big(\bar{\boldsymbol{w}}_*\big)_j$ is the j-th entry of $\bar{\boldsymbol{w}}_* = Q\boldsymbol{w}_*$.

Proof: We proceed in steps.

Convergence in the mean:

The error equation for the leaky LMS algorithm is

$$\tilde{\boldsymbol{w}}_{k+1}^l = \big((1-\mu\alpha_0)I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k^l - \mu\boldsymbol{x}_k v(k) + \mu\alpha_0\boldsymbol{w}_*. \tag{2.39}$$

Taking expectations, and using I-1, M-1, we obtain

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1}^l = \big((1-\mu\alpha_0)I - \mu R\big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l + \mu\alpha_0\boldsymbol{w}_*,$$

and the result follows directly. In particular, under the theorem's conditions, $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ converges exponentially fast to its limit.

Convergence in the mean-square:

The recursion for the matrix $\bar{C}_k = \mathrm{E}\,\tilde{\boldsymbol{w}}_k^l \tilde{\boldsymbol{w}}_k^{l,T}$ is given by

$$\begin{aligned}\bar{C}_{k+1} = (1-\mu\alpha_0)^2\bar{C}_k - \mu(1-\mu\alpha_0)\big(\bar{C}_k R + R\bar{C}_k\big) + \mu^2 R\bar{C}_k R + \mu^2 R\,\mathrm{Tr}\big(R\bar{C}_k\big) - \\ - \mu\alpha_0\big(I - \mu\alpha_0 I - \mu R\big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l \boldsymbol{w}_*^T - \mu\alpha_0 \boldsymbol{w}_*\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k^{l,T}\big(I - \mu\alpha_0 I - \mu R\big) + \\ + \mu^2\alpha_0^2 \boldsymbol{w}_*\boldsymbol{w}_*^T + \mu^2\sigma_v^2 R.\end{aligned}$$

Let again Q be an orthogonal matrix that diagonalizes R, i.e.,

$$\Lambda = Q^T R Q, \qquad\qquad Q^T Q = I,$$

and define $C_k = Q^T \bar{C} Q$. Then the recursion for C_k is

$$\begin{aligned}C_{k+1} = (1-\mu\alpha_0)^2 C_k - \mu(1-\mu\alpha_0)\big(C_k\Lambda + \Lambda C_k\big) + \mu^2\Lambda C_k\Lambda + \mu^2\Lambda\,\mathrm{Tr}\big(\Lambda C_k\big) - \\ - \mu\alpha_0\big(I - \mu\alpha_0 I - \mu\Lambda\big)\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l \bar{\boldsymbol{w}}_*^T - \mu\alpha_0 \bar{\boldsymbol{w}}_*\,\mathrm{E}\,\tilde{\boldsymbol{w}}_k^{l,T}\big(I - \mu\alpha_0 I - \mu\Lambda\big) + \\ + \mu^2\alpha_0^2 \bar{\boldsymbol{w}}_*\bar{\boldsymbol{w}}_*^T + \mu^2\sigma_v^2\Lambda.\end{aligned}$$

Note that the homogeneous part of this recursion (i.e., removing the constant terms *and* the $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ terms) is similar to that for the LMS algorithm. Using arguments very similar to those of Theorem 2.2, it can be shown that the origin $C = 0$ for this homogeneous recursion is exponentially stable. Using this fact, and since we have shown that $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ converges exponentially fast to its limit, we can replace $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ by $\mathrm{E}\,\tilde{\boldsymbol{w}}_\infty^l$ to study the steady-state behavior of the leaky LMS algorithm. Since again the arguments are not much different than those used in the proof of Theorem 2.2, we shall not repeat them here — see [Say97].

Note that Assumption R-1 is not necessary for this proof.

♢

We should point out that the $\bar{C}_\infty$ computed above is not the true covariance of $\tilde{\boldsymbol{w}}_\infty^l$, even asymptotically — to obtain the true covariance, we should subtract

$$\left(\mathrm{E}\,\tilde{\boldsymbol{w}}_\infty^l\right)\left(\mathrm{E}\,\tilde{\boldsymbol{w}}_\infty^l\right)^T.$$

We did not compute the variance, since the error of interest in applications is really $e(k)$.

2.3.2 Stationary Non-Gaussian Inputs

When the input sequence is not Gaussian, the following new result can be established. The assumptions are as before, but with G-1 substituted by:

I-3. *The noise sequence* $\{v(k)\}$ *is independent of the input sequence,*

IS-2. *The fourth-order moments of* $\boldsymbol{x}_k$ *are bounded by a constant* B*, i.e.,*

$$\mathrm{E}\left(\boldsymbol{x}_k^T\boldsymbol{x}_k\right)^2 < B < \infty.$$

Theorem 2.7. *Assume that I-1–I-4, IS-1 and IS-2 hold. Under these conditions, if the step-size satisfies* (2.38)*, the weight error vector* $\tilde{\boldsymbol{w}}_k^l$ *computed by the leaky LMS algorithm converges in the mean to*

$$\lim_{k\to\infty}\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l = \alpha_0(\alpha_0 I + R)^{-1}\boldsymbol{w}_*. \tag{2.40}$$

In addition, there is a $\mu_0 > 0$ *such that, for all* $\mu < \mu_0$*, the steady-state MSE is*

$$\lim_{k\to\infty}\mathrm{E}\,e(k)^2 \approx \sigma_v^2 + \sum_{i=1}^{M}\frac{\mu\sigma_v^2\lambda_i^2}{2(\alpha_0+\lambda_i)} + \sum_{i=1}^{M}\frac{\alpha_0^2\left(\bar{\boldsymbol{w}}_*\right)^2}{\lambda_i(\alpha_0+\lambda_i)^2}, \tag{2.41}$$

where $\left(\bar{\boldsymbol{w}}_*\right)_i$ *is as in the previous theorem.*

If the fourth-order moments of $\boldsymbol{x}_k$ are known, the matrix $\bar{C}_k = \mathrm{E}\,\tilde{\boldsymbol{w}}_k^l \tilde{\boldsymbol{w}}_k^{l,T}$ can be computed from the recursion:

$$\begin{bmatrix} \mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1}^l \\ \mathrm{vec}(\bar{C}_{k+1}) \end{bmatrix} = \begin{bmatrix} (1-\mu\alpha_0)I - \mu R & 0 \\ G_1 & G_2 \end{bmatrix} \begin{bmatrix} \mathrm{E}\,\tilde{\boldsymbol{w}}_k^l \\ \mathrm{vec}(\bar{C}_k) \end{bmatrix} + \\ + \begin{bmatrix} \mu\alpha_0 \boldsymbol{w}_* \\ \mu^2\alpha_0^2\,\mathrm{vec}(\boldsymbol{w}_*\boldsymbol{w}_*^T) + \mu^2\sigma_v^2\,\mathrm{vec}(R) \end{bmatrix}, \tag{2.42}$$

where G_1 and G_2 are given by

$$G_1 \triangleq -\mu\alpha_0 \boldsymbol{w}_* \otimes \big((1-\mu\alpha_0)I - \mu R\big) - \mu\alpha_0\big((1-\mu\alpha_0)I - \mu R\big) \otimes \boldsymbol{w}_*,$$

$$G_2 \triangleq \Big[(1-\mu\alpha_0)^2 I_{M^2} - \mu(1-\mu\alpha_0)\big(R \otimes I_M\big) - \mu(1-\mu\alpha_0)\big(I_M \otimes R\big) + \\ + \mu^2\,\mathrm{E}\,\big(\boldsymbol{x}_k\boldsymbol{x}_k^T \otimes \boldsymbol{x}_k\boldsymbol{x}_k^T\big)\Big],$$

and the initial condition is

$$\begin{bmatrix} \mathrm{E}\,\tilde{\boldsymbol{w}}_0^l \\ \mathrm{vec}\big(\mathrm{E}\,\tilde{\boldsymbol{w}}_0^l \tilde{\boldsymbol{w}}_0^{l\,T}\big) \end{bmatrix}.$$

Proof: The argument here follows closely the proofs of Theorems 2.3 and 2.4. The main difference is that, since the recursion for $\bar{C}_k$ depends on $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$, we need to consider the recursion for $\bar{C}_k$ jointly with that for the mean, in order to obtain an exact model.

In fact, the recursion for the mean in (2.42) is independent from $\mathrm{vec}\big(\bar{C}_k\big)$, and thus under the conditions of the theorem, $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ will converge exponentially fast to the limit in (2.40).

On the other hand, since the matrix

$$A \triangleq \begin{bmatrix} (1-\mu\alpha_0)I - \mu R & 0 \\ G_1 & G_2 \end{bmatrix}$$

is block-lower triangular, the leaky LMS algorithm will be stable if, and only if,

$$-1 < \lambda\Big((1-\mu\alpha_0)I - \mu R\Big) < 1, \qquad \text{and} -1 \qquad < \lambda(G_2) < 1.$$

Again we can follow a procedure similar to the proof of Thm. 2.3 to show that for small enough μ the term $\mu^2\,\mathrm{E}\left(\boldsymbol{x}_k\boldsymbol{x}_k^T \otimes \boldsymbol{x}_k\boldsymbol{x}_k^T\right)$ can be ignored.

We conclude that leaky LMS computes stable (in the mean-square sense) estimates under the conditions of the theorem. Moreover, in steady-state we have (already ignoring higher powers of μ and simplifying),

$$\Big[2\alpha_0 I_{M^2} + R\otimes I_M + I_M \otimes R\Big]\,\mathrm{vec}\big(\bar{C}_\infty\big) \approx$$

$$\approx \left[\alpha_0\left(\mu\alpha_0\boldsymbol{w}_*\otimes I_M + \mu\alpha_0 I_M\otimes\boldsymbol{w}_*\right)(\alpha_0 I+R)^{-1}\boldsymbol{w}_* + \mu\alpha_0^2\,\mathrm{vec}(\boldsymbol{w}_*\boldsymbol{w}_*^T) + \mu\sigma_v^2\,\mathrm{vec}(R)\right].$$

Applying (2.22) to this relation, we obtain

$$2\alpha_0\bar{C}_\infty + R\bar{C}_\infty + \bar{C}_\infty R \approx$$

$$\mu\alpha_0^2\boldsymbol{w}_*\boldsymbol{w}_*^T(\alpha_0 I+R)^{-1} + \mu\alpha_0^2(\alpha_0 I+R)^{-1}\boldsymbol{w}_*\boldsymbol{w}_*^T + \mu\alpha_0^2\boldsymbol{w}_*\boldsymbol{w}_*^T + \mu\sigma_v^2 R.$$

Let Q be an orthogonal matrix that diagonalizes R, as in (2.2). Multiplying the above relation from the left by Q and from the right by Q^T, and defining $\bar{\boldsymbol{w}}_* \triangleq Q\boldsymbol{w}_*$, the simplified relation below is obtained,

$$2\alpha_0 C_\infty + \Lambda C_\infty + C_\infty\Lambda \approx \alpha_0^2\bar{\boldsymbol{w}}_*\bar{\boldsymbol{w}}_*^T(\alpha_0 I+\Lambda)^{-1} + \alpha_0^2(\alpha_0 I+\Lambda)^{-1}\bar{\boldsymbol{w}}_*\bar{\boldsymbol{w}}_*^T +$$
$$+\,\mu\alpha_0^2\bar{\boldsymbol{w}}_*\bar{\boldsymbol{w}}_*^T + \mu\sigma_v^2\Lambda.$$

Note that the term $\mu\alpha_0^2\boldsymbol{w}_*\boldsymbol{w}_*^T$ can be ignored when compared to

$$\alpha_0^2\bar{\boldsymbol{w}}_*\bar{\boldsymbol{w}}_*^T(\alpha_0 I+\Lambda)^{-1}.$$

Moreover, only the diagonal entries of C_∞ enter into the expressions for both the MSE and the MSD. Each diagonal entry can be computed independently from

the others from the above relation. Performing the computations, after a little algebra we obtain (2.41).

$\diamondsuit$

2.4 INDEPENDENCE THEORY FOR NLMS

This section provides a new precise analysis of the performance of the NLMS algorithm

$$\boldsymbol{w}_{k+1}^n = \boldsymbol{w}_k^n + \frac{\mu}{a + \|\boldsymbol{x}_k\|^2}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k^n\big).$$

The method will lead to less restrictive conclusions than available in the literature (e.g., [Slo93]). Our derivation is based on the following observation. Define the new variables

$$\boldsymbol{x}_k^{(n)} \triangleq \frac{\boldsymbol{x}_k}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}, \qquad v^{(n)}(k) \triangleq \frac{v(k)}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}, \tag{2.43a}$$

and

$$y^{(n)}(k) \triangleq \frac{y(k)}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}, \tag{2.43b}$$

and define the new covariance matrix

$$R^{(n)} \triangleq \mathrm{E}\, \boldsymbol{x}_k^{(n)} \boldsymbol{x}_k^{(n)\,T}. \tag{2.44}$$

With these new variables, the normalized NLMS algorithm is equivalent to the LMS algorithm applied to the model

$$y^{(n)}(k) = \boldsymbol{x}_k^{(n)\,T} \boldsymbol{w}_* + v^{(n)}(k), \qquad \text{with} \qquad e^{(n)}(k) = \boldsymbol{x}_k^{(n)\,T} \tilde{\boldsymbol{w}}_k^n + v^{(n)}(k).$$

Note that if Assumption I-3 holds, $v^{(n)}(k)$ and $\boldsymbol{x}_k$ are still *uncorrelated*, but not necessarily independent. This relation between the LMS and the NLMS algorithms holds for any input distribution, either stationary or non-stationary. The

relation can be used in general to easily translate results for the LMS algorithm to the NLMS algorithm.

Although this change of variables has been used before (see, e.g., [WL90, ABH97]), the analyses available in the literature are lacking due to three main reasons:

1. Only the case $a = 0$ is considered (the case $a \neq 0$ is common in practice, since it avoids potential numerical problems when $\boldsymbol{x}_k = \mathbf{0}$).

2. The conditions for stability are based on results valid for LMS with Gaussian variables — but it is clear that $\boldsymbol{x}_k^{(n)}$ cannot be Gaussian (since it is bounded).

3. No attention was given to the crucial fact that $\boldsymbol{x}_k^{(n)}$ and $v^{(n)}(k)$ are still uncorrelated when Assumption I-3 holds. In this case, we can easily show that the NLMS algorithm still computes unbiased estimates.

In this section, we resolve these issues and study the performance of the NLMS algorithm under the following assumptions.

M-1. *The sequences* $\{y(k), \boldsymbol{x}_k\}$ *are related through a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2*.*

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-3. *The noise sequence* $\{v(k)\}$ *is independent of the input sequence.*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

R-1. *The matrix R is positive-definite ($R > 0$).*

IS-1. *The random sequences $\{\boldsymbol{x}_k\}$ and $\{y(k)\}$ are jointly stationary.*

Note that the transformed input sequence $\{\boldsymbol{x}_k^n\}$ cannot be Gaussian (since it is bounded), therefore the new result below is for general input distributions.

Theorem 2.8. *Under the above assumptions, the NLMS algorithm computes asymptotically unbiased estimates and converges in the mean-square sense if $0 < \mu < 2$ (for any input distribution). The MSE is, in steady-state and for $\mu \approx 0$,*

$$J \approx \sigma_v^2 + \frac{\mu}{2}\sigma_{xv}^2, \tag{2.45}$$

where

$$\sigma_{xv}^2 \triangleq \sigma_v^2 \operatorname{Tr}\left\{ R R^{(n)^{-1}} \operatorname{E}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\left(a + \|\boldsymbol{x}_k\|^2\right)^2} \right) \right\},$$

and the MSD is

$$\mathcal{D} \approx \frac{\mu}{2}\sigma_v^2 \operatorname{Tr}\left\{ R^{(n)^{-1}} \operatorname{E}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\left(a + \|\boldsymbol{x}_k\|^2\right)^2} \right) \right\}. \tag{2.46}$$

Proof: First note that Assumption R-1 implies that $R^{(n)} > 0$. Indeed, assume that $R^{(n)}$ is singular, i.e., that there exists a vector $\boldsymbol{u}$ such that

$$\boldsymbol{u}^T R^{(n)} \boldsymbol{u} = 0.$$

Therefore, by a sequence of arguments similar to those used to derive (2.34), we conclude that the set

$$\mathcal{C} \triangleq \left\{ \boldsymbol{x} : \ \boldsymbol{u}^T \frac{\boldsymbol{x}}{\sqrt{a + \|\boldsymbol{x}\|^2}} \neq 0 \right\}$$

has probability 0. It follows that the event $\boldsymbol{u}^T \boldsymbol{x} = 0$ has probability 1, which implies that R is singular. This proves that R is singular if $R^{(n)}$ is singular. A

similar set of arguments shows that the converse is also true. We conclude that $R^{(n)}$ is positive-definite if, and only if, R is positive-definite.

Convergence in the mean: The NLMS error equation in the new variables is

$$\tilde{\boldsymbol{w}}_{k+1}^n = \left(I - \boldsymbol{x}_k^{(n)} \boldsymbol{x}_k^{(n)\,T}\right) \tilde{\boldsymbol{w}}_k^n - \mu \boldsymbol{x}_k^{(n)} v^{(n)}(k).$$

Taking expected values, and recalling that $v^{(n)}(k)$ and $\boldsymbol{x}_k^{(n)}$ are uncorrelated, we obtain

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1}^n = \left(I - R^{(n)}\right) \mathrm{E}\,\tilde{\boldsymbol{w}}_k^n,$$

which converges to zero if, and only if,

$$R^{(n)} > 0, \qquad\qquad \mu\lambda_{\max}\left(R^{(n)}\right) < 2.$$

The first of the above conditions is satisfied if, and only if, $R > 0$, as we have just proved. The second quantity is satisfied if $\mu < 2$, since

$$\lambda_{\max}\left(\boldsymbol{x}_k^{(n)} \boldsymbol{x}_k^{(n)\,T}\right) = \lambda_{\max}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{a + \|\boldsymbol{x}_k\|^2}\right) \leq 1$$

implies that the covariance $R^{(n)}$ satisfies

$$\begin{aligned}\lambda_{\max}(R^{(n)}) &= \max_{\|\boldsymbol{u}\|=1} \boldsymbol{u}^T \left[\mathrm{E}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{a + \|\boldsymbol{x}_k\|^2}\right)\right] \boldsymbol{u} = \\ &= \max_{\|\boldsymbol{u}\|=1} \mathrm{E}\left(\boldsymbol{u}^T \frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{a + \|\boldsymbol{x}_k\|^2} \boldsymbol{u}\right) \leq 1. \end{aligned} \tag{2.47}$$

Convergence in the mean-square: Although the results for LMS can be translated with few modifications to the NLMS algorithm using our change of variables, in the stability proof we can say more if we do not do so. Indeed, since the input sequence is bounded, it can be shown that the covariance matrix $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^n \tilde{\boldsymbol{w}}_k^{n,T}$ converges to steady-state if $0 < \mu < 2$, for *any* input distribution. This is as

opposed to the LMS algorithm, where stability conditions are highly dependent on the input distribution (through $\lambda_{\max}(R)$).

Therefore, the proof we present below is significantly different than that for the LMS algorithm (and, in fact, simpler). Only when we compute the steady-state MSE, we shall use the results from Theorem 2.3.

Define the matrix of fourth-order moments

$$R_4^{(n)} \triangleq \mathrm{E}\big(\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\big)^2. \tag{2.48}$$

The key remark in the stability proof is that $R_4 \leq R$, no matter what the distribution of $\boldsymbol{x}_k$ is. To prove this, let $\boldsymbol{\alpha}$ be any constant vector. The argument below shows that $\boldsymbol{\alpha}^T(R_4 - R)\boldsymbol{\alpha} \leq 0$ for all $\boldsymbol{\alpha}$:

$$\begin{aligned}\boldsymbol{\alpha}^T\big(\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\big)^2\boldsymbol{\alpha} &= \boldsymbol{\alpha}^T\boldsymbol{x}_k^{(n)}\big(\boldsymbol{x}_k^{(n)\,T}\boldsymbol{x}_k^{(n)}\big)\boldsymbol{x}_k^{(n)\,T}\boldsymbol{\alpha} = \\ &= \frac{\|\boldsymbol{x}_k\|^2}{a+\|\boldsymbol{x}\|^2}\boldsymbol{\alpha}^T\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\boldsymbol{\alpha} \leq \boldsymbol{\alpha}^T\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\boldsymbol{\alpha},\end{aligned} \tag{2.49}$$

since $\|\boldsymbol{x}_k\|/\big(a+\|\boldsymbol{x}_k\|^2\big) \leq 1$. Since the inequalities hold for all $\boldsymbol{x}_k^{(n)}$, it follows that

$$\boldsymbol{\alpha}^T R_4^{(n)}\boldsymbol{\alpha}^T = \mathrm{E}\left(\boldsymbol{\alpha}^T\big(\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\big)^2\boldsymbol{\alpha}\right) \leq \mathrm{E}\big(\boldsymbol{\alpha}^T\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\boldsymbol{\alpha}\big) = \boldsymbol{\alpha}^T R^{(n)}\boldsymbol{\alpha}. \tag{2.50}$$

Now assume that I-3 and I-4 hold, and determine the recursion for $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2$:

$$\mathrm{E}\,\|\tilde{\boldsymbol{w}}_{k+1}^n\|^2 = \mathrm{E}\left(\tilde{\boldsymbol{w}}_k^{n,T}\big(I-\mu\boldsymbol{x}_k^{(n)}\boldsymbol{x}_k^{(n)\,T}\big)^2\tilde{\boldsymbol{w}}_k^n\right) + \mu^2\,\mathrm{E}\big(\|\boldsymbol{x}_k^{(n)}\|^2 v^{(n)}(k)^2\big),$$

where the independence of $\boldsymbol{x}_k$ and $v(k)$ has been used to cancel the cross-terms. Expand this expression and use I-1 to separate the expectations of $\tilde{\boldsymbol{w}}_k^n$ and of $\boldsymbol{x}_k^{(n)}$:

$$\mathrm{E}\,\|\tilde{\boldsymbol{w}}_{k+1}^n\|^2 = \mathrm{E}\left(\tilde{\boldsymbol{w}}_k^{n,T}\big(I-2\mu R^{(n)}+\mu^2 R_4^{(n)}\big)\tilde{\boldsymbol{w}}_k^n\right) + \mu^2\,\mathrm{E}\big(\|\boldsymbol{x}_k^{(n)}\|^2 v^{(n)}(k)^2\big).$$

Let $Q^{(n)}$ be an orthogonal matrix that diagonalizes $R^{(n)}$,

$$\operatorname{diag}(\lambda_i^{(n)}) \triangleq \Lambda^{(n)} \triangleq Q^{(n)\,T} R^{(n)} Q^{(n)},$$

and define

$$\boldsymbol{\omega}_k \triangleq Q^{(n)\,T} \tilde{\boldsymbol{w}}_k^n, \qquad \boldsymbol{\xi}_k \triangleq Q^{(n)\,T} \boldsymbol{x}_k^{(n)} = Q^{(n)\,T} \frac{\boldsymbol{x}_k}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}. \tag{2.51}$$

Using the new variables, (2.50), and noting that $\|Q^{(n)\,T} \boldsymbol{x}_k\| = \|\boldsymbol{x}_k\|$, the recursion for $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2$ becomes

$$\begin{aligned} \mathrm{E}\,\|\tilde{\boldsymbol{w}}_{k+1}^n\|^2 = \mathrm{E}\,\|\boldsymbol{\omega}_{k+1}\|^2 = \boldsymbol{\omega}_k^T \big(I - 2\mu\Lambda^{(n)} - \mu^2 Q^{(n)\,T} R_4 Q^{(n)}\big)\boldsymbol{\omega}_k + \\ + \mu^2\,\mathrm{E}\big(\boldsymbol{x}_k^{(n)\,T} \boldsymbol{x}_k^{(n)} v^{(n)}(k)^2\big). \end{aligned} \tag{2.52}$$

Let $\boldsymbol{e}_i$ be the i-th canonical vector, and use (2.50) to conclude that

$$\boldsymbol{e}_i^T Q^{(n)\,T} R_4 Q^{(n)} \boldsymbol{e}_i \leq \boldsymbol{e}_i^T \Lambda^{(n)} \boldsymbol{e}_i = \lambda_i^{(n)}.$$

Using this result, we obtain

$$\boldsymbol{\omega}_k^T \big(2\Lambda^{(n)} - \mu Q^{(n)\,T} R_4 Q^{(n)}\big)\boldsymbol{\omega}_k \leq \sum_{i=0}^{M} \lambda_i^{(n)} (2-\mu) \big(\boldsymbol{\omega}_k\big)_i,$$

where $\big(\boldsymbol{\omega}_k\big)_i$ is the i-th entry of the vector $\boldsymbol{\omega}_k$. Applying this result to the recursion for $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2$, we obtain

$$\begin{aligned} \mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2 &\leq \sum_{i=1}^{M} \left(\big(1 - \mu(2-\mu)\lambda_i^{(n)}\big) \big(\boldsymbol{\omega}_k\big)_i^2 \right) + \mu^2\,\mathrm{E}\big(\|\boldsymbol{x}_k^{(n)}\|^2 v^{(n)}(k)^2\big) \\ &\leq \big(1 - \mu(2-\mu)\lambda_{\min}(R^{(n)})\big)\,\mathrm{E}\,\|\boldsymbol{\omega}_k\|^2 + \mu^2\,\mathrm{E}\big(\|\boldsymbol{x}_k^{(n)}\|^2 v^{(n)}(k)^2\big). \end{aligned}$$

From this recursion, and remembering that $\lambda_i^{(n)} \leq 1$ and that $\|\boldsymbol{w}_k^n\| = \|\boldsymbol{\omega}_k\|$, it follows that $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2$ is bounded if $0 < \mu < 2$.

If $\mu \approx 0$, the results from Theorem 2.3 can be used to predict the steady-state MSE. The recursion for the diagonal elements of the covariance matrix of $\boldsymbol{\omega}_k$ is

obtained by simply copying equation (2.13). Ignoring fourth-order moments, the recursion is

$$\Omega_{k+1} \stackrel{\Delta}{=} \operatorname{diag}\big(\mathrm{E}\,\boldsymbol{\omega}_{k+1}\boldsymbol{\omega}_{k+1}^T\big) \approx \big(I - 2\mu\Lambda^{(n)}\big)\Omega_k + \mu^2 D_k,$$

where D_k is

$$D_k \stackrel{\Delta}{=} \mathrm{E}\big(\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T v^{(n)}(k)^2\big) = \sigma_v^2\,\mathrm{E}\left(\frac{\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T}{a + \|\boldsymbol{x}_k\|^2}\right).$$

The steady-state Ω_k is then

$$\Omega_\infty = \lim_{k\to\infty}\Omega_k = \mu\big(2\Lambda^{(n)}\big)^{-1}D.$$

Note that the goal is to compute $\mathrm{E}\,e(k)^2$, not $\mathrm{E}\,e^{(n)}(k)^2$. Therefore the steady-state MSE for NLMS is

$$\lim_{k\to\infty}\mathrm{E}\,e(k)^2 = \sigma_v^2 + \operatorname{Tr}\big(\Lambda\Omega_\infty\big) = \sigma_v^2 + \mu\frac{1}{2}\operatorname{Tr}\big(\Lambda\Lambda^{(n)^{-1}}D\big).$$

Defining

$$\sigma_{xv}^2 = \operatorname{Tr}\mathrm{E}\big(\Lambda\Lambda^{(n)^{-1}}\boldsymbol{\xi}_k^T\boldsymbol{\xi}_k v^{(n)}(k)^2\big) = \sigma_v^2\operatorname{Tr}\left\{\Lambda\Lambda^{(n)^{-1}}\,\mathrm{E}\left(\frac{\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T}{a + \|\boldsymbol{x}_k\|^2}\right)\right\},$$

the steady-state MSE is

$$\lim_{k\to\infty}\mathrm{E}\,e(k)^2 = \sigma_v^2 + \frac{\mu}{2}\sigma_{xv}^2. \tag{2.53}$$

The value of σ_{xv}^2 can also be obtained experimentally.

$\diamondsuit$

Note that if $a = 0$ in the NLMS recursion (1.10), the stability condition becomes both necessary and sufficient. Indeed, when $a = 0$,

$$\|\boldsymbol{x}_k^{(n)}\| = 1,$$

and therefore,

$$R_4^{(n)} = \mathrm{E}\left(\boldsymbol{x}_k^{(n)} \left(\boldsymbol{x}_k^{(n)\,T} \boldsymbol{x}_k^{(n)} \right) \boldsymbol{x}_k^{(n)\,T} \right) = R^{(n)}.$$

Using this relation, the recursion (2.52) for $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k^n\|^2$ becomes

$$\mathrm{E}\,\|\tilde{\boldsymbol{w}}_{k+1}^n\|^2 = \sum_{i=1}^{M} \left(\left(1 - \mu(2-\mu)\lambda_i^{(n)}\right) \left(\boldsymbol{\omega}_k\right)_i^2 \right) + \mu^2\,\mathrm{E}\big(\|\boldsymbol{x}_k^{(n)}\|^2 v^{(n)}(k)^2\big),$$

and since $\lambda_i^{(n)} \leq 1$, this recursion converges if and only if $0 < \mu < 2$.

Appendix 2.C describes an approximate model that has been suggested in the literature [Slo93] to compute the MSE and the convergence rate for $a = 0$. The results we derived reduce to the approximate values of the appendix if it is assumed that

$$\mathrm{E}\left(\frac{\boldsymbol{\xi}_k \boldsymbol{\xi}_k^T}{\|\boldsymbol{x}_k\|^2} \right) \approx \left(\mathrm{E}\,\boldsymbol{\xi}_k \boldsymbol{\xi}_k^T \right) \left(\mathrm{E}\,\frac{1}{\|\boldsymbol{x}_k\|^2} \right) = \Lambda^{(n)} \left(\mathrm{E}\,\frac{1}{\|\boldsymbol{x}_k\|^2} \right).$$

Inserting this approximation into the expression for σ_{xv}^2 we obtain, for a small step-size,

$$\lim_{k\to\infty} \mathrm{E}\,e(k)^2 \approx \sigma_v^2 + \mu \frac{\sigma_v^2}{2} \mathrm{Tr}\big(\Lambda\big)\,\mathrm{E}\left(\frac{1}{\|\boldsymbol{x}_k\|^2} \right),$$

which is exactly the expression in the appendix when $\mu \ll 2$. Hence, our results are less restrictive.

2.5 CONTRIBUTIONS OF THIS CHAPTER

We reviewed the main stability and performance results for adaptive filters when the inputs and noise form stationary independent sequences. We also established several new results, in particular the change of variables for normalized LMS (Sec. 2.4 and Thm. 2.8); Theorem 2.7, which studies the behavior of the leaky

LMS algorithms with non-Gaussian inputs; and the analysis of LMS with singular input covariance matrix in Sec. 2.2 and Thm. 2.5.

Although the analysis of adaptive filters using the independence assumptions has been used for over three decades now [WMG67], it was only until [FW85] that a precise study of LMS with independent Gaussian $\{\boldsymbol{x}_k\}$ was developed. Similarly, the first precise analysis of leaky LMS with Gaussian and independent inputs appeared only in [MA97, Say97].

While the case of singular input-covariance matrices was considered in [Lin88], we extended the results, providing an expression for the MSD, and eliminating some unnecessary assumptions.

Precise analyses of the NLMS algorithm with Gaussian, independent $\{\boldsymbol{x}_k\}$ appeared in [TF88], and a simplified model (different than the one presented here, see Appendix 2.C) for small step-size and $a = 0$ was proposed in [Slo93].

References [WL90, ABH97] use the change of variables described in Sec. 2.4, but their analyses fail in three respects:

1. Only the case $a = 0$ is considered (the case $a \neq 0$ is perhaps more common in practice, since it avoids potential numerical problems when $\boldsymbol{x}_k = \mathbf{0}$).

2. The conditions for stability are based on results valid for LMS with Gaussian variables — but it is clear that $\boldsymbol{x}_k^{(n)}$ (as defined in Eq. (2.43)) cannot be Gaussian (since it is bounded).

3. No attention was given to the crucial fact that $\boldsymbol{x}_k^{(n)}$ and $v^{(n)}(k)$ are still uncorrelated when Assumption I-3 holds. In this case, we have shown that the NLMS algorithm still computes unbiased estimates, while references [WL90, ABH97] claim that the estimates are biased.

Our analysis not only resolves these three points, but is also valid for a generic input distribution (i.e., it allows for non-Gaussian $\boldsymbol{x}_k$).

APPENDICES FOR CHAPTER 2

2.A SOME USEFUL RESULTS

2.A.1 A Result Concerning Gaussian Variables

Lemma 2.A.1. *Let $\boldsymbol{x}$ be a real random variable normally distributed with zero mean and diagonal covariance matrix Λ. For any matrix Π (symmetric or not), the following equality holds.*

$$\mathrm{E}\Big\{\boldsymbol{x}\boldsymbol{x}^T\Pi\boldsymbol{x}\boldsymbol{x}^T\Big\} = \Lambda\,\mathrm{Tr}\big(\Pi\Lambda\big) + \Lambda\Pi\Lambda + \Lambda\Pi^T\Lambda.$$

Proof: The proof is based on the fact that uncorrelated Gaussian variables are also independent, so if $x_{(i)}$ is the i-th element of $\boldsymbol{x}$, then $x_{(i)}$ is independent of $x_{(j)}$ for $i \neq j$. Let K denote the matrix we are looking for, and K_{ij} be its i,j-th element. Then

$$K_{ij} = \mathrm{E}\left\{x_{(i)}x_{(j)}\left(\sum_{m=1}^{M} x_{(m)}\right)\left(\sum_{n=1}^{M}\Pi_{mn}x_{(n)}\right)\right\}.$$

The right-hand side is nonzero only when there are two pairs of equal indices $\{i=j,\, m=n\}$ or $\{i=m,\, j=n\}$ or $\{i=n,\, j=m\}$. Assume first that $i=j$ (diagonal elements). Then the expectation is nonzero only for $m=n$, i.e.

$$K_{ij} = \mathrm{E}\left\{x_{(i)}^2\sum_m \Pi_{mm}x_{(m)}^2\right\} = \sum_m \Pi_{mm}\,\mathrm{E}\big\{x_{(i)}^2x_{(m)}^2\big\} = \lambda_i\,\mathrm{Tr}\big(\Pi\Lambda\big) + 2\Pi_{ii}\lambda_i^2,$$

where we used $\mathrm{E}\,x^4 = 3\sigma_x^4$, if $x \sim N(0,\sigma_x^2)$.

For the off-diagonal elements ($i \neq j$), we must have either $i=n$, $j=m$, or

$i = m$, $j = n$, so

$$K_{ij} = \mathrm{E}\left\{ x_{(i)} \sum_{m=1}^{M} x_{(m)} \Pi_{mj} x_j^2 \right\} + \mathrm{E}\left\{ x_{(j)} \sum_{n=1}^{M} x_{(n)} \Pi_{in} x_i^2 \right\} =$$
$$= \Big(\Pi_{ij} + \Pi_{ji} \Big)\, \mathrm{E}\Big\{ x_{(i)}^2 x_{(j)}^2 \Big\} = \Big(\Pi_{ij} + \Pi_{ji} \Big) \lambda_i \lambda_j .$$

$\diamondsuit$

The equivalent result for complex variables is the following.

Lemma 2.A.2. *Let $\boldsymbol{z}$ be a complex random variable normally distributed with zero mean and diagonal covariance matrix Λ, and assume that $\mathrm{E}\,\boldsymbol{z}\boldsymbol{z}^T = 0$ (if $\boldsymbol{z} = \boldsymbol{x} + j\boldsymbol{y}$, this implies that $\mathrm{E}\,\boldsymbol{x}\boldsymbol{x}^T = \mathrm{E}\,\boldsymbol{y}\boldsymbol{y}^T$).*

Under these conditions,

$$\mathrm{E}\Big\{ \boldsymbol{x}\boldsymbol{x}^H \boldsymbol{x}\boldsymbol{x}^H \Big\} = \Lambda\, \mathrm{Tr}\big(\Lambda\big) + \Lambda^2 .$$

Proof: Almost the same as the proof for the previous lemma. The only difference is that the fourth moment of a complex circular (and scalar) variable $z = x + jy$ with zero mean and variance σ^2 is

$$\mathrm{E}\,|z|^4 = \mathrm{E}\big[x^2 + y^2\big]^2 = 2\sigma^4 . \tag{2.A.1}$$

$\diamondsuit$

2.A.2 Matrix Inversion Lemma

Assume that A and C are invertible. Then the following equality holds [Kai80]

$$\big(A + BCD\big)^{-1} = A^{-1} - A^{-1}B\big(C^{-1} + DA^{-1}B\big)^{-1}DA^{-1} .$$

2.B INDEPENDENCE THEORY IN FINITE-PRECISION ARITHMETIC

In this appendix we study the behavior of the LMS algorithm in finite-precision arithmetic, using the independence assumptions and also assuming stationarity. The results presented here were used in Sec. 2.2, and will also be used in Theorems 6.2 and 6.3 in Chapter 6.

Our presentation is based on the results by [CL84] and [Ale87]; see also [Cio87, BB96b, BB96a]. Note that we still assume independence of $\boldsymbol{x}_k$ (as all the works treating finite-precision arithmetic to date do).

We assume that the computations are carried using fixed-point arithmetic with rounding. The data variables ($y(k)$ and $\boldsymbol{x}_k$) are stored with B_d bits plus sign, while the weight estimates $\boldsymbol{w}_k$ are stored using B_c bits plus sign. This results in multiplication rounding errors with variance $\sigma_d^2 = \frac{2^{-2B_d}}{12}$ for data, and $\sigma_c^2 = \frac{2^{-2B_c}}{12}$ for weights.

The weight estimates computed by the FX LMS algorithm will be denoted $\boldsymbol{z}_k$, and the weight error, $\tilde{\boldsymbol{z}}_k$. We will write $\bar{\boldsymbol{x}}_k$ for the quantized version of $\boldsymbol{x}_k$, $\bar{e}(k) = \text{fx}[\bar{y}(k) - \boldsymbol{z}_k^T \bar{\boldsymbol{x}}_k]$. The symbol fx$[a]$ denotes the fixed-point representation of a.

In addition to assumptions I-1–I-2 and G-1, we need the following:

IFX-1. *All variables are scaled so that overflow never occurs (this is a standard assumption for fixed-point, and in fact amounts to assuming that all variables are bounded).*

IFX-2. $\text{Q}[\boldsymbol{x}_k] = \bar{\boldsymbol{x}}_k = \boldsymbol{x}_k + \boldsymbol{\alpha}_k$, $\text{Q}[y] = \bar{y}(k) = y(k) + \beta(k)$, *and the sequences* $\{\boldsymbol{\alpha}_k\}$ *and* $\{\beta(k)\}$ *are iid, independent of all other variables (including* $\boldsymbol{x}_k$ *and*

$y(k)$), and with covariances $\sigma_d^2 I$ and σ_d^2, respectively.

IFX-3. *The error $\eta(k)$, defined by* $\mathrm{fx}[\boldsymbol{z}_k^T \bar{\boldsymbol{x}}_k] = \boldsymbol{z}_k^T \boldsymbol{x}_k + \boldsymbol{z}_k^T \boldsymbol{\alpha}_k + \eta(k)$, *forms an iid sequence independent of all other variables. Its variance σ_η^2 is dependent on how the inner product is computed. If each multiplication is rounded before the additions, then $\sigma_\eta^2 = M\sigma_d^2$. Otherwise, if the quantization is made only after the additions, $\sigma_\eta^2 = \sigma_d^2$. We define C such that*

$$\sigma_\eta^2 = C\sigma_d^2,$$

i.e., $C = 1$ or $C = M$, depending on how the inner product is computed.

Similarly, the error $\boldsymbol{\xi}_k$ defined by $\mathrm{fx}[\mu \bar{\boldsymbol{x}}_k \bar{e}(k)] = \mu \bar{\boldsymbol{x}}_k \bar{e}(k) + \boldsymbol{\xi}_k$ *is iid, independent of all other errors, and has covariance Σ_ξ.*

The vector $\boldsymbol{\xi}_k$ depends on how the product $\mu \bar{\boldsymbol{x}}_k \bar{e}(k)$ is computed. If μ is a power of two, then $\boldsymbol{\xi}_k$ has covariance $\Sigma_\xi = \sigma_c^2 I$. Otherwise,

$$\mathrm{fx}[\mu \bar{\boldsymbol{x}}_k \bar{e}(k)] = \mathrm{fx}\Big[\mathrm{fx}[\mu \bar{e}(k)] \bar{\boldsymbol{x}}_k\Big] = \big(\mu \bar{e}(k) + \xi_k^{'}\big) \bar{\boldsymbol{x}}_k + \boldsymbol{\xi}_k^{''},$$

where $\xi_k^{'}$ (a scalar) and $\boldsymbol{\xi}_k^{''}$ (a vector) are independent, so the total covariance of $\boldsymbol{\xi}_k = \boldsymbol{\xi}_k^{''} + \xi_k^{'} \bar{\boldsymbol{x}}_k$ is

$$\Sigma_\xi \stackrel{\Delta}{=} \sigma_c^2 \big((1 + \sigma_d^2) I + R\big),$$

(where we assumed that the product $\mu \bar{e}(k)$ is stored with B_c bits).

IFX-4. *All quantization errors η satisfy*

$$\mathrm{E}\,\|\eta\|^p \approx 0 \quad \textit{for } p \geq 3.$$

This is a reasonable approximation if $\sigma_d^2 \ll \lambda_i(R)$, and given the fact that the arithmetic errors are bounded by $2^{-B_d - 1} M$.

The theorem below is a known result, and was first proved in [CL84] (though there is an oversight in their argument). The proof given here is different (and more straightforward).

Theorem 2.B.1 (Steady-state error). *Under assumptions R-1, I-1–I-4, IS-1–IS-2 and IFX-1–IFX-3, the steady-state error of the FX LMS filter is (for small μ)*

$$\begin{aligned}\lim_{k\to\infty} \mathrm{E}\,\bar{e}^2(k) = \sigma_v^2 + \mu\frac{\sigma_v^2}{2}\operatorname{Tr} R + \frac{\mu\sigma_d^2 \operatorname{Tr} R}{2}\Big(\|\boldsymbol{w}_*\|^2 + (1+C)\Big) + \\ + \frac{\operatorname{Tr}\big(\Sigma_\xi\big)}{2\mu} + (1+C)\sigma_d^2 + \frac{\mu\sigma_d^2\sigma_v^2 M}{2}\end{aligned} \tag{2.B.1}$$

where Σ_ξ depends on how $\mathrm{Q}[\mu \boldsymbol{x}' e'(k)]$ is computed.

Proof: Define $e(k) = \tilde{\boldsymbol{z}}_k^T \boldsymbol{x}_k + v(k)$. The recursion for the FX LMS algorithm is

$$\begin{aligned}\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \mu e(k)\boldsymbol{x}_k + \mu e(k)\boldsymbol{\alpha}_k + \mu\big(\beta(k) - \eta(k)\big)\boldsymbol{x}_k + \\ + \mu\big(\beta(k) - \eta(k)\big)\boldsymbol{\alpha}_k - \mu\boldsymbol{z}_k^T\boldsymbol{\alpha}_k\boldsymbol{x}_k - \mu\boldsymbol{\alpha}_k^T\boldsymbol{z}_k\boldsymbol{\alpha}_k + \boldsymbol{\xi}_k,\end{aligned}$$

and the error equation becomes

$$\begin{aligned}\tilde{\boldsymbol{z}}_{k+1} = \Big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T - \mu\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T\Big)\tilde{\boldsymbol{z}}_k - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\xi}_k - \\ - \mu\big(\beta(k) - \eta(k)\big)\boldsymbol{\alpha}_k + \mu\boldsymbol{w}_*^T\boldsymbol{\alpha}_k\boldsymbol{x}_k - \mu\tilde{\boldsymbol{z}}_k^T\boldsymbol{\alpha}_k\boldsymbol{x}_k - \\ - \mu\tilde{\boldsymbol{z}}_k^T\boldsymbol{x}_k\boldsymbol{\alpha}_k - \mu\boldsymbol{\alpha}_k v(k) - \mu\big(\beta(k) - \eta(k)\big)\boldsymbol{x}_k + \mu\boldsymbol{\alpha}_k^T\boldsymbol{w}_*\boldsymbol{\alpha}_k.\end{aligned} \tag{2.B.2}$$

Taking expectations and using the independence assumptions, we obtain

$$\mathrm{E}\,\tilde{\boldsymbol{z}}_{k+1} = \Big(I - \mu R - \mu\sigma_d^2 I\Big)\,\mathrm{E}\,\tilde{\boldsymbol{z}}_k + \mu\sigma_d^2\boldsymbol{w}_*,$$

from which we conclude that the expected value of the error converges to

$$\lim_{k\to\infty}\mathrm{E}\,\tilde{\boldsymbol{z}}_k = \sigma_d^2\big(R + \sigma_d^2 I\big)^{-1}\boldsymbol{w}_*,$$

if $0 < \mu(\lambda_i + \sigma_d^2) < 2$, where λ_i are the eigenvalues of R. Note that the bias is nonzero only if $\boldsymbol{\alpha}_k$ is nonzero.

Define the covariance matrix $Z_k = Q\,\mathrm{E}\,\tilde{\boldsymbol{z}}_k\tilde{\boldsymbol{z}}_k^T Q^T$, where the orthogonal matrix Q diagonalizes R, as before. Evaluating $Q\,\mathrm{E}\,\tilde{\boldsymbol{z}}_{k+1}\tilde{\boldsymbol{z}}_{k+1}^T Q^T$, we recognize that the terms with Z_k are similar to the terms with C_k in (2.11), only with

$$\Lambda' = \Lambda + \sigma_d^2 I$$

in place of Λ. In the relation below we already canceled the cross-terms that average to zero, and invoked assumption IS-2 to ignore fourth-order moments (we denote $Q\Sigma_\xi Q^T = \Lambda_\xi$, a diagonal matrix)

$$\begin{aligned} Z_{k+1} = Z_k - \mu\big(Z_k\Lambda' + \Lambda' Z_k\big) + O\big(\mu^2 Z_k\big) + \mu^2\sigma_v^2\Lambda' + \\ + \Lambda_\xi + \mu^2(1+C)\sigma_d^2\Lambda + \mu^2\sigma_d^2\|\boldsymbol{w}_*\|^2\Lambda. \end{aligned} \tag{2.B.3}$$

This recursion is very similar to (2.11), but with $\Lambda_\xi + \mu^2(1+C)\sigma_d^2\Lambda + \mu^2\sigma_d^2\Lambda\|\boldsymbol{w}_*\|^2$ added to the constant term and the modified Λ'. We conclude that the conditions for convergence are similar to those in Theorem 2.3.

The steady-state value of Z_k is found using the results of the previous section. Define

$$\begin{aligned} \boldsymbol{L}' &= \begin{bmatrix} \lambda_1 + \sigma_d^2 & \lambda_2 + \sigma_d^2 & \dots & \lambda_M + \sigma_d^2 \end{bmatrix}^T, \\ \boldsymbol{L}_\xi &= \mathrm{diag}(\Lambda_\xi), \end{aligned}$$

then the diagonal of Z_k is

$$\lim_{k\to\infty} \mathrm{diag}(Z_k) \approx \big(2\mu\Lambda'\big)^{-1}\Big(\mu^2\sigma_v^2\boldsymbol{L}' + \boldsymbol{L}_\xi + \mu^2(1+C)\sigma_d^2\boldsymbol{L} + \mu^2\sigma_d^2\|\boldsymbol{w}_*\|^2\boldsymbol{L}\Big), \tag{2.B.4}$$

and (2.B.1) follows.

♢

2.C ALTERNATIVE INDEPENDENCE ANALYSIS FOR NLMS

An approximate method to simplify the analysis of the NLMS algorithm has been proposed in [Slo93]. Unlike our results in Sec. 2.4, this approximation only works for the NLMS variant for $a = 0$, viz.,

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \frac{\boldsymbol{x}_k}{\|\boldsymbol{x}_k\|^2}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k\big). \tag{2.C.1}$$

The idea is to *choose* a probability distribution for $\boldsymbol{x}_k$ that simplifies the analysis; it is chosen so that its first and second order moments are the same as those of $\boldsymbol{x}_k$. Since for small μ, the misadjustment and convergence rate depend primarily on these low order moments, the results obtained using this alternative distribution should be reasonable.

The assumptions used here are:

M-1. *The sequences* $\big\{y(k),\, \boldsymbol{x}_k\big\}$ *are related through a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2*.*

I-1. *The sequence* $\big\{\boldsymbol{x}_k\big\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-3. *The noise sequence* $\big\{v(k)\big\}$ *is independent of the input sequence.*

I-4. *The noise sequence* $\big\{v(k)\big\}$ *is independent and identically distributed (iid).*

R-1. *The matrix* R *is positive-definite* ($R > 0$)*.*

IS-1. *The random sequences* $\big\{\boldsymbol{x}_k\big\}$ *and* $\big\{y(k)\big\}$ *are jointly stationary.*

The last assumption defines the input distribution. Let again Q be an orthogonal matrix that diagonalizes R (see (2.2)). Denote the i-th column of Q by $\boldsymbol{q}_i$ so that

$$R = Q\Lambda Q^T = \sum_{i=1}^{M} \lambda_i \boldsymbol{q}_i \boldsymbol{q}_i^T.$$

IS-2. *Assume that*

$$\boldsymbol{x}_k = s(k) r(k) \boldsymbol{V}_k, \tag{2.C.2}$$

where the random variables $r(k)$, $s(k)$ and $\boldsymbol{V}_k$ are independent, $r(k)$ has the same probability distribution as $\|\boldsymbol{x}_k\|$, and

$$\begin{aligned} \mathrm{P}\big\{s(k) = \pm 1\big\} &= \frac{1}{2}, \\ \mathrm{P}\big\{\boldsymbol{V}_k = \boldsymbol{q}_i\big\} &= p_i = \frac{\lambda_i}{\operatorname{Tr} R}, \quad i = 1 \ldots M. \end{aligned}$$

Even though this assumption is rarely met in practice, it does simplify the analysis. In addition, for small step-sizes only the second-order statistics of $\boldsymbol{x}_k$ and $v(k)$ are important for the analysis (as we shall see in Sec. 3.1). Note that the covariance of $\boldsymbol{V}_k$ is

$$\mathrm{E}\, \boldsymbol{V}_k \boldsymbol{V}_k^T = \frac{1}{\operatorname{Tr} R} \sum_{i=1}^{M} \lambda_i \boldsymbol{q}_i \boldsymbol{q}_i^T = \frac{R}{\operatorname{Tr} R}.$$

Define the covariance matrix $\bar{C}_k = \mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T$ as before. From the NLMS recursion (2.C.1), it follows that

$$\begin{aligned} \bar{C}_{k+1} &= \mathrm{E}\left(\Big[I - \mu \frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\|\boldsymbol{x}_k\|^2}\Big] \bar{C}_k \Big[I - \mu \frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\|\boldsymbol{x}_k\|^2}\Big] \right) + \mu^2 \sigma_v^2 \left(\mathrm{E}\, \frac{1}{r^2} \right) \mathrm{E}\, \boldsymbol{V}\boldsymbol{V}^T = \\ &= \mathrm{E}\left(\Big[I - \mu \frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\|\boldsymbol{x}_k\|^2}\Big] \bar{C}_k \Big[I - \mu \frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\|\boldsymbol{x}_k\|^2}\Big] \right) + \frac{\mu^2 \sigma_v^2}{\operatorname{Tr} R} R \left(\mathrm{E}\, \frac{1}{r^2} \right) \end{aligned} \tag{2.C.3}$$

Instead of solving this recursion, recall that $\mathrm{E}\, e^2(k) = \operatorname{Tr}\big(R\bar{C}_k\big)$ and define

$$\tilde{\lambda}_i(k) = \Big(Q^T \bar{C}_k Q\Big)_{ii},$$

where $\big(A\big)_{ii}$ is the $(i,\, i)$-th element of A. With this definition,

$$\mathrm{E}\, e(k)^2 = \sum_{i=1}^{M} \lambda_i \tilde{\lambda}_i(k).$$

A recursion for $\tilde{\lambda}_i(k)$ can be obtained by pre- and post-multiplying (2.C.3) by $\boldsymbol{q}_i^T$ and $\boldsymbol{q}_i$, respectively. Performing the multiplications and noting that

$$\boldsymbol{q}_i^T \frac{\boldsymbol{x}_k}{\|\boldsymbol{x}_k\|} = \begin{cases} 0 & \text{with probability } 1 - p_i, \\ 1 & \text{w.p. } p_i, \end{cases}$$

a recursion for $\tilde{\lambda}$ follows,

$$\tilde{\lambda}_i(k+1) = \left(1 - \mu(2-\mu)\frac{\lambda_i}{\mathrm{Tr}\, R}\right)\tilde{\lambda}_i(k) + \frac{\mu^2\sigma_v^2}{\mathrm{Tr}\, R}\lambda_i \left(\mathrm{E}\,\frac{1}{r(k)^2}\right). \qquad (2.\mathrm{C}.4)$$

This recursion converges if $0 < \mu < 2$. The steady-state value is

$$\lim_{k\to\infty} \tilde{\lambda}_i(k) = \frac{\mu^2\sigma_v^2}{\mu(2-\mu)\lambda_i}\left(\mathrm{E}\,\frac{1}{r(k)^2}\right),$$

from which the steady-state MSE can be computed.

$$\lim_{k\to\infty} \mathrm{E}\, e(k)^2 = \frac{\mu\sigma_v^2}{2-\mu}\left(\mathrm{E}\,\frac{1}{r(k)^2}\right)\mathrm{Tr}\, R.$$

The value of $\mathrm{E}\,\frac{1}{r(k)^2}$ cannot, in general, be obtained from knowledge of R alone.

2.D NON-STATIONARY GAUSSIAN INPUTS

The tracking performance of the LMS algorithm with independent and Gaussian $\{\boldsymbol{x}_k\}$ is well-known. We summarize here the main result. We assume again that G-1 holds (i.e., that $\{y(k),\, \boldsymbol{x}_k\}$ are jointly Gaussian). However, we are now interested in the tracking capability of the LMS algorithm when the model assumed in M-1 is in fact time-variant. Hence we now assume the following conditions:

M-2. *The sequences* $\{y(k),\ \boldsymbol{x}_k\}$ *are related through a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_{*,\,k} + v(k)$$

for some unknown time-variant sequence $\{\boldsymbol{w}_{*,\,k}\}$*, and where* $v(k)$ *is again zero-mean with variance* σ_v^2 *and uncorrelated with* $\boldsymbol{x}_k$*.*

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

R-1. *The matrix* R *is positive-definite* ($R > 0$)*.*

G-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly Gaussian.*

In addition, we also need to assume IV-1, IV-2, and IV-3 as explained below. Condition IV-1 is

IV-1. *The sequences* $\{\boldsymbol{x}_k\}$, $\{v(k)\}$ *and* $\{\boldsymbol{w}_{*,\,k}\}$ *are statistically independent.*

A bound on how fast the "true" weight vectors $\{\boldsymbol{w}_{*,\,k}\}$ change with time is also necessary. There are several ways to do this. The assumption presented here models $\boldsymbol{w}_{*,\,k}$ as a stochastic process. It is also possible to model $\boldsymbol{w}_{*,\,k}$ as a deterministic, but time-variant sequence (see Sec. 3.2).

IV-2. $\boldsymbol{w}_{*,\,k}$ *is a stationary first order autoregressive process, i.e., it is generated by a model of the form*

$$\begin{aligned} \boldsymbol{w}_{f,\,k+1} &= \theta \boldsymbol{w}_{f,\,k} + \eta_k, \qquad |\theta| < 1, \\ \boldsymbol{w}_{*,\,k} &= \boldsymbol{w}_* + \boldsymbol{w}_{f,\,k}, \end{aligned} \tag{2.D.1}$$

where η_k *is a zero mean Gaussian white noise of variance* $\sigma_\eta^2 I$*, and* $\boldsymbol{w}_{f,\,0}$ *is a random variable with zero mean, and* $\boldsymbol{w}_*$ *is a constant vector.*

The next assumption is not necessary, but it simplifies the analysis. Results relaxing this assumption will be presented in Thm. 3.A.1, in Sec. 3.A.

IV-3. *The sequences* $\{\boldsymbol{x}_k\}$ *and* $\{v(k)\}$ *are wide-sense stationary (i.e., they have constant first and second-order moments).*

With IV-3, there is only one difference between the results for this time-variant case and the stationary case. The misadjustment, M_d, will no longer be proportional to the step-size μ. The conditions for stability and convergence do not change.

Theorem 2.D.1 (Convergence in the mean). *The expected value of* $\boldsymbol{w}_k$ *converges exponentially to* $\boldsymbol{w}_*$ *(the "DC" part of* $\boldsymbol{w}_{*,\,k}$*) if Assumptions R-1, I-1–I-2, G-1 and IV-1–IV-2 hold, and if, and only if, the step-size* μ *satisfies*

$$\mu < \frac{2}{\lambda_{\max}\left(\mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T\right)}. \tag{2.D.2}$$

Proof: We write the error equation, simultaneously with the recursion for $\boldsymbol{\Delta}_k \triangleq \boldsymbol{w}_{f,\,k+1} - \boldsymbol{w}_{f,\,k}$, as

$$\begin{aligned} \tilde{\boldsymbol{w}}_{k+1} &= \left(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\right)\tilde{\boldsymbol{w}}_k + \mu\boldsymbol{x}_k v(k) - \boldsymbol{\Delta}_k, \\ \boldsymbol{\Delta}_{k+1} &= \theta\boldsymbol{\Delta}_{f,k} + \boldsymbol{\eta}_{k+1} - \boldsymbol{\eta}_k. \end{aligned} \tag{2.D.3}$$

Taking expectations, recursions similar to those for the stationary case are obtained:

$$\begin{bmatrix} \mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1} \\ \mathrm{E}\,\boldsymbol{\Delta}_{k+1} \end{bmatrix} = \begin{bmatrix} I - \mu R & I \\ 0 & \theta I \end{bmatrix} \begin{bmatrix} \mathrm{E}\,\tilde{\boldsymbol{w}}_k \\ \mathrm{E}\,\boldsymbol{\Delta}_k \end{bmatrix}$$

The expected values of $\tilde{\boldsymbol{w}}_k$ and $\boldsymbol{\Delta}_k$ tend to zero under the assumptions of the theorem. Note that if $\boldsymbol{w}_{f,\,0}$ is assumed to have nonzero mean (or is deterministic),

the proof does not change.

♢

The analysis for the covariance is similar. The proof for the following theorem follows [SK95, pp. 124–126].

Theorem 2.D.2 (Steady-state MSE). *Under the Assumptions R-1, I-1–I-4 and IV-1–IV-3, the covariance matrix* $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$ *will reach a steady-state if, and only if,* $\mu\lambda_i < 1$ *and* $0 < c < 1$*, where* c *is as defined in* (2.4)*. In addition, the misadjustment will be*

$$M_d = \frac{c}{1-c} + \frac{\sigma_\eta^2}{\sigma_v^2}\frac{d}{1-c}, \tag{2.D.4}$$

and

$$d \triangleq \frac{1}{1+\theta}\sum_{i=1}^{M}\frac{\lambda_i}{(1-\theta+\mu\theta\lambda_i)(1-\mu\lambda_i)}. \tag{2.D.5}$$

Proof: Defining

$$\boldsymbol{\omega}_k \triangleq \begin{bmatrix}\tilde{\boldsymbol{w}}_k^T & \tilde{\boldsymbol{w}}_{f,k}^T\end{bmatrix}^T, \qquad \xi_k \triangleq \begin{bmatrix}\boldsymbol{x}_k^T & \boldsymbol{0}^T\end{bmatrix}^T,$$
$$B \triangleq \begin{bmatrix}I & (1-\theta)I\\ 0 & \theta I\end{bmatrix}, \qquad \boldsymbol{\psi}_k \triangleq \begin{bmatrix}\mu\boldsymbol{x}_k v(k) - \boldsymbol{\eta}_k\\ \boldsymbol{\eta}_k\end{bmatrix},$$

the time-variant LMS equations (2.D.3) can be rewritten as

$$\boldsymbol{\omega}_{k+1} = \left(B - \mu\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T\right)\boldsymbol{\omega}_k + \boldsymbol{\psi}_k.$$

The covariances of $\boldsymbol{\xi}_k$ and $\boldsymbol{\psi}_k$ are given by

$$R_\xi \triangleq \mathrm{E}\,\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T = \begin{bmatrix}R & 0\\ 0 & 0\end{bmatrix}, \qquad R_\psi \triangleq \mathrm{E}\,\boldsymbol{\psi}_k\boldsymbol{\psi}_k^T = \begin{bmatrix}I & -I\\ -I & I\end{bmatrix},$$

and therefore the covariance $\bar{\Omega}_k \triangleq \mathrm{E}\,\boldsymbol{\omega}_k\boldsymbol{\omega}_k^T$ satisfies the recursion

$$\begin{aligned}\bar{\Omega}_{k+1} = B\bar{\Omega}_k B^T - \mu\left(B\bar{\Omega}_k R_\xi + R_\xi\bar{\Omega}_k B^T\right) + 2\mu^2 R_\xi\bar{\Omega}_k R_\xi + \\ + \mu^2\,\mathrm{Tr}(\bar{\Omega}_k R_\xi)R_\xi + \mu^2\sigma_v^2 R_\xi + \sigma_\eta^2 R_\psi.\end{aligned} \tag{2.D.6}$$

Let again Q be an orthogonal matrix such that $Q^T R Q = \Lambda$, and define

$$S \triangleq \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix}, \qquad \text{and} \qquad \Omega_k \triangleq S^T\bar{\Omega}_k S.$$

Rewriting the covariance recursion (2.D.6) in terms of Ω_k, we obtain

$$\begin{aligned}\Omega_{k+1} = B\Omega_k B^T - \mu\left(B\Omega_k\Lambda_\xi + \Lambda_\xi\Omega_k B^T\right) + 2\mu^2\Lambda_\xi\Omega_k\Lambda_\xi + \\ + \mu^2\,\mathrm{Tr}(\Omega_k\Lambda_\xi)R_\xi + \mu^2\sigma_v^2\Lambda_\xi + \sigma_\eta^2 R_\psi,\end{aligned}$$

where

$$\Lambda_\xi \triangleq \begin{bmatrix} \Lambda & 0 \\ 0 & 0 \end{bmatrix}.$$

This recursion can be de-coupled if we divide Ω_k into blocks

$$\Omega_k = \begin{bmatrix} \Omega_{k,1} & \Omega_{k,2}^T \\ \Omega_{k,2} & \Omega k,3 \end{bmatrix},$$

and write recursions for each of these blocks separately. Starting with $\Omega_{k,3}$, we obtain

$$\Omega_{k+1,3} = \theta^2\Omega_{k,3} + \sigma_\eta^2 I,$$

from which we conclude that

$$\lim_{k\to\infty}\Omega_{k,3} = \sigma_\eta^2(1-\theta^2)^{-1}I,$$

if, and only if, $-1 < \theta < 1$.

On the other hand, $\Omega_{k,2}$ satisfies

$$\Omega_{k+1,2} = \theta\Omega_{k,2}(I - \mu\Lambda) + \theta(1-\theta)\Omega_{k,3} - \sigma_\eta^2 I.$$

This recursion will converge if, and only if,

$$|\theta(1 - \mu\lambda_i)| < 1,$$

where λ_i are the eigenvalues of R. Under this condition, the steady-state value will be

$$\lim_{k\to\infty} \Omega_{k,2} = -\sigma_\eta^2 \frac{1}{1+\theta} \left((1-\theta)I + \mu\theta\Lambda\right)^{-1}.$$

Finally, the recursion for $\Omega_{k,1}$ is

$$\begin{aligned}\Omega_{k+1,1} = \Omega_{k,1} &- \mu\Omega_{k,1}\Lambda - \mu\Lambda\Omega_{k,1} + 2\mu^2\Lambda\Omega_{k,1}\Lambda + \\ &+ (1-\theta)\left(\Omega_{k,2} + \Omega_{k,2}^T\right) + (1-\theta)^2\Omega_{k,3} - \mu(1-\theta)\left(\Omega_{k,2}\Lambda + \Lambda\Omega_{k,2}\right) + \\ &+ \mu^2\left(\mathrm{Tr}(\Omega_{k,1}\Lambda) + \sigma_v^2\right)\Lambda + \sigma_\eta^2 I.\end{aligned} \tag{2.D.7}$$

Note that $\Omega_{k,2}$ and $\Omega_{k,3}$ can be regarded as time-functions in the above recursion, since neither of them depends on $\Omega_{k,1}$. This implies that, to study the stability of this recursion, the terms in $\Omega_{k,2}$ and $\Omega_{k,3}$ can be ignored. The homogeneous recursion for $\Omega_{k,1}$ is therefore,

$$\Omega_{k+1,1} = \Omega_{k,1} - \mu\Omega_{k,1}\Lambda - \mu\Lambda\Omega_{k,1} + 2\mu^2\Lambda\Omega_{k,1}\Lambda + \mu^2\,\mathrm{Tr}(\Omega_{k,1}\Lambda).$$

Comparing the above recursion with (2.9), we conclude that the stability conditions are as in Theorem 2.2.

To find the steady-state $\Omega_{\infty,1}$, replace $\Omega_{\infty,2}$ and $\Omega_{\infty,3}$ into (2.D.7), and solve for $\Omega\infty, 1$. Eq. (2.D.4) is obtained from the expression $\mathrm{E}\, e(\infty)^2 = \mathrm{Tr}(R\bar{C}_\infty)$.

$\diamondsuit$

CHAPTER 3

THE SMALL STEP-SIZE CASE

This chapter summarizes several results from the literature concerning the analysis of the LMS algorithm (1.6) for small step-sizes μ and without the independence assumptions. Basically, it is shown that if $\mu \approx 0$, the conclusions obtained using the independence assumptions are good approximations for the performance of the algorithm. Most of the results presented here are for the LMS algorithm. Some results in Secs. 3.2 and 3.3 also apply for the normalized LMS algorithm. The proofs of the theorems described in this chapter are very lengthy and are not important for the next chapters. For this reason they are omitted, but references to the literature are given.

There are no extensions of most of these results for the leaky-LMS algorithm. The reason for this gap is that the methods are either especially tailored for LMS (as in Secs. 3.1 and 3.2), or require that the origin $\tilde{\boldsymbol{w}}_k^l = \mathbf{0}$ of the algorithm's error equation be an exponentially-stable equilibrium point when the noise is identically zero, which is not the case for leaky LMS. It may be possible to extend the results of Secs. 3.1 and 3.2 to leaky LMS, but the analysis would be extremely lengthy and may not be justified. Instead of trying to do so, we shall present in Chapter 6 a new leaky algorithm, *circular-leaky*, that avoids most of the drawbacks of leaky LMS (namely, the new algorithm computes unbiased estimates and has smaller computational cost than leaky LMS). For this new algorithm, the results of this chapter can be applied. See Chapter 6 and [NS96, NS99c].

Some of the results presented here for the LMS algorithm will be extended for the NLMS algorithm by using the change of variables described in Sec. 2.4.

3.1 PERFORMANCE RESULTS FOR LMS

This section summarizes results of [Maz79, JCR82] which, in a sense, justify the use of the independence assumptions.

The following assumption describes the basic model in [Maz79] (the "NDS" stands for *N-dependent and stationary*).

NDS-1. *The input sequence* $\{\boldsymbol{x}_k\}$ *is created from a tap-delay line, say*

$$\boldsymbol{x}_k = \begin{bmatrix} a(k-M+1) & a(k-M+2) & \dots & a(k) \end{bmatrix}^T. \tag{3.1}$$

The sequence $\{a(k)\}$ *is in turn the output of a linear time-invariant FIR filter* $\mathcal{H}$*,*

$$a(k) = \sum_{i=0}^{N-1} h(i)s(k-i), \tag{3.2}$$

where $\{s(k)\}$ *is a* binary *iid sequence and* $\{h(i)\}$ *are the filter coefficients (see Fig. 3.1).*

The result in [Maz79] assumes that the input sequence $\{s(k)\}$ is *binary*, i.e., $s(k) = \pm 1$. This model is adequate for equalization applications with BPSK modulation. Without significant change in the arguments, the assumption of a binary input sequence $\{s(k)\}$ can be relaxed to allow for a sequence which takes a finite (power of 2) number of values.

Using the binary $s(k)$ model, it is shown in [Maz79] that the results obtained from independence theory can be interpreted as an approximation of the actual performance of LMS. The main result is the following.

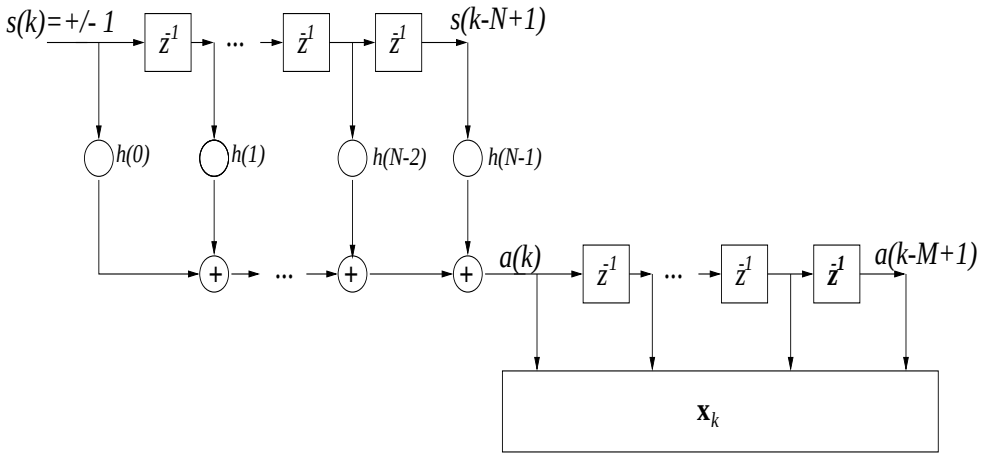

Figure 3.1: *Model for the input sequence* $\{\boldsymbol{x}_k\}$ *in [Maz79].*

Theorem 3.1. *Assume that the input sequence is generated as in NDS-1 with binary* $s(k)$*, and that assumptions I-4. IS-1 and R-1 hold, i.e.,*

I-4. *The noise sequence* $\{v(k)\}$ *is independent and identically distributed (iid).*

R-1. *The matrix* R *is positive-definite* ($R > 0$).

IS-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

Then the error between the true value of the MSE and the approximation given by Theorem 2.3 in (2.23) *is of the order of* μ^2*, that is,*

$$\lim_{k\to\infty} \mathrm{E}\, e(k)^2 = \sigma_v^2 + \frac{\mu \operatorname{Tr} R}{2}\sigma_v^2 + O(\mu^2).$$

In addition, the rate of convergence of the average weight error vector $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$ *given by Theorem 2.1 is also correct up to order of* μ. ♢

A generalization of this result that allows for input sequences $\{s(k)\}$ that do not come from a finite alphabet is found in [JCR82]. Instead of NDS-1, the input

is assumed to satisfy:

NDS-2. *Let $a(k) = s(k) * h(k)$, where $\{s(k)\}$ is an iid sequence, and $h(k)$ is the impulse response of a causal, length-N FIR filter ($h(k) = 0$ for $k < 0$ and $k \geq N$). Assume that the input sequence is*

$$\left\{ \boldsymbol{x}_k = \begin{bmatrix} a(k-M+1) & a(k-M+2) & \dots & a(k) \end{bmatrix}^T \right\}$$

The following theorem extends Theorem 3.1 to the above class of inputs, and is proved in [JCR82].

Theorem 3.2 (N-dependent steady-state). *If Assumptions R-1, IS-1, IS-2 and NDS-2 hold, and if the covariance $\bar{C}_k = \mathrm{E}\,\tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T$ converges to a constant, then the steady-state values $\mathrm{E}\,\tilde{\boldsymbol{w}}_\infty$, $\bar{C}_\infty$ and $\mathrm{E}\,e^2(\infty)$, computed by the LMS algorithm, can be expanded as power series in μ as below:*

$$\lim_{k\to\infty} \mathrm{E}\,\tilde{\boldsymbol{w}}_k = \mu R^{-1} \sum_{i=1}^{N_0-1} \mathrm{E}\big(\boldsymbol{x}_i \boldsymbol{x}_i^T \boldsymbol{x}_0 v(0)\big) + O\big(\mu^2\big), \tag{3.3}$$

$$\lim_{k\to\infty} \mathrm{E}\,e^2(k) = \sigma_v^2 + \mu \left[\frac{1}{2} \operatorname{Tr} \mathrm{E}\Big(v^2(0)\boldsymbol{x}_0\boldsymbol{x}_0^T\Big) - \sum_{i=1}^{2N_0-1} \operatorname{Tr} \mathrm{E}\Big(v(0)v(i)\boldsymbol{x}_0\boldsymbol{x}_i^T\Big) \right] + \\ + O\big(\mu^2\big), \tag{3.4}$$

$$\lim_{k\to\infty} \bar{C}_k = \mu \bar{C}_\infty^{(1)} + O\big(\mu^2\big), \tag{3.5}$$

where

$$R\bar{C}_\infty^{(1)} + \bar{C}_\infty^{(1)} R = \sum_{i=1-2N_0}^{2N_0-1} \mathrm{E}\Big(v(0)v(i)\boldsymbol{x}_0\boldsymbol{x}_i^T\Big),$$

In addition, if $\mu \approx 0$, the mean $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$ converges to its limit approximately as a

linear system with modes $1 - \mu\lambda_i$*, where* λ_i *are the eigenvalues of* R*.*

$\diamondsuit$

Note that it is *not* assumed that the noise sequence is iid or independent of the input. If $\{v(k)\}$ is iid and independent of $\{\boldsymbol{x}_k\}$, the above equations reduce to the results predicted by independence theory. Another remark is that there is no proof of convergence for $\bar{C}_k$ in [Maz79, JCR82], *the convergence is assumed.* The convergence (or stability) proof can be found in [ME83], whose results we summarize next.

3.2 CONVERGENCE RESULTS FOR LMS

The most important characteristic of the model in NDS-2 is that two input vectors $\boldsymbol{x}_k$ and $\boldsymbol{x}_j$ are independent if the time difference $|k-j|$ exceeds the sum $M+N$ of the lengths of $\boldsymbol{x}_k$ and of the modelling filter $\mathcal{H}$. This means that $\tilde{\boldsymbol{w}}_k$ in LMS is independent of all input vectors $\boldsymbol{x}_j$ for which $j-k > N$. This fact is the basis of the convergence proof in [ME83].

For the convergence analysis, the assumption NDS-2 can be relaxed to

NDS-3. *The extended sequence* $\{y(k),\, \boldsymbol{x}_k\}$ *is such that*

$$\{\ldots,\, (y(k-2),\, \boldsymbol{x}_{k-2}),\, (y(k-1),\, \boldsymbol{x}_{k-1}),\, (y(k),\, \boldsymbol{x}_k)\} \qquad \textit{and}$$
$$\{(y(k+j),\, \boldsymbol{x}_{k+j}),\, (y(k+j+1),\, \boldsymbol{x}_{k+j+1}),\, \ldots\}$$

are mutually independent whenever $j > N$*. Sequences satisfying this property are said to be* N-dependent.

In addition, it is necessary to bound the moments of the input and desired

sequences in the following way.

NDS-4. $\mathrm{E}\left(\|\boldsymbol{x}_k\|^{2nN}\right) < \infty$, *for all* $n \leq 12$.

NDS-5. $\mathrm{E}\left(|y(k)|^{4+\frac{4}{6N-1}}\right) < \infty$.

Theorem 3.3. *If Assumptions R-1, IS-1 and NDS-2–NDS-5 hold with* $N > M/12$, *then there exists a pair* (μ_0, β) *of positive real numbers such that the weight error vector* $\tilde{\boldsymbol{w}}_k$ *given by the LMS algorithm* (1.6) *satisfies*

$$\limsup_{k\to\infty} \mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2 \leq \beta\mu, \quad \forall\, \mu \leq \mu_0.$$

$\diamondsuit$

The values of the constants β and μ_0 are not easily obtainable from the proof in [ME83]. In particular, β depends on $\mathrm{E}\left(\lambda_{\min}\left(\sum_{i=0}^{M-1} \boldsymbol{x}_{k+iN}\boldsymbol{x}_{k+iN}^T\right)\right)$.

There is an extension of this result for non-stationary input sequences [EM85]. We summarize these results in Appendix 3.A.

3.3 AVERAGING ANALYSIS

Similar performance results were derived in [Sol89, Sol92, Sol94b] using a different set of assumptions, and by using a different method of analysis (averaging) that applies to a wider class of algorithms. The input sequence $\{\boldsymbol{x}_k\}$ is now not required to be N-dependent anymore — instead, it is required to satisfy a mixing condition (i.e., the correlation of $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ "dies out" as the time difference $|m-n|$ increases). In addition, $\{\boldsymbol{x}_k\}$ is also required to be bounded, i.e., we assume that there exists a $B_x < \infty$ such that

B-1.

$$\sup_{k>0} \boldsymbol{x}_k^T \boldsymbol{x}_k \leq B_x < \frac{2}{\mu}, \quad \textit{with probability } 1.$$

Note that the condition is not satisfied in the important case of LMS with Gaussian-distributed input sequences. However, assumption B-1 is *always* satisfied for the normalized LMS algorithm, since the transformed input $\boldsymbol{x}_k^{(n)}$ from Sec. 2.2 is always bounded. Moreover, *averaging theory* [SK95, BMP87, Kus84, KY97] requires that the step-size be vanishingly small. Nevertheless, and unlike the previous results (Thms. 3.2 and 3.3), this theory can be applied to a larger class of adaptive algorithms, with little modification in the basic theorems. In particular, we shall use Thm. 3.4 to analyze our new algorithm in Sec. 6.6. The drawback, as we mentioned above, is that important input distributions (such as Gaussian) are ruled out.

To describe the correlation properties of the sequence $\{\boldsymbol{x}_k\}$, we need the following definition.

Definition 1 (uniform-mixing processes). A random process $\boldsymbol{\xi}_k$ is called a uniform- (or ϕ-) mixing process if there exists a sequence $\phi(n)$ satisfying

$$\phi(n) \xrightarrow[n\to\infty]{} 0,$$

such that

$$\sup_{A\in\mathcal{F}_{t+s}^{\infty},\, B\in\mathcal{F}_0^t} \big|\mathrm{P}(A|B) - \mathrm{P}(A)\big| \leq \phi(s), \quad \forall\, t, s, \tag{3.6}$$

where $\mathcal{F}_t^s \triangleq \sigma\{\boldsymbol{\xi}_u,\ t \leq u \leq s\}$ (The symbol $\sigma\{\cdot\}$ denotes the smallest sigma-algebra generated by a set [Dur96]).

This condition says that two variables $\boldsymbol{\xi}_u$ and $\boldsymbol{\xi}_s$ become essentially independent as the time difference $|u-s|$ grows.

Examples of ϕ-mixing processes are deterministic processes, N-dependent processes, and processes generated from bounded white noise filtered through a stable finite-dimensional linear filter [Guo94].

We then assume that

UM-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is uniform mixing with mixing function* $\phi(n)$.

We shall list here the main theorems of general averaging theory, and later specialize the results to the LMS and NLMS algorithms.

3.3.1 General Averaging Analysis

Consider an adaptive update of the general form

$$\tilde{\boldsymbol{w}}_{k+1} = \tilde{\boldsymbol{w}}_k + \mu f(k,\, \tilde{\boldsymbol{w}}_k), \quad \text{with some initial condition } \tilde{\boldsymbol{w}}_0, \tag{3.7}$$

where $\tilde{\boldsymbol{w}}_k$ is the error vector we want to minimize. The function f is stochastic, *i.e.,* for every k and $\tilde{\boldsymbol{w}}_k$, $f(k,\, \tilde{\boldsymbol{w}}_k)$ is a random vector. We could be more explicit in the notation and write $f(\boldsymbol{\xi}_k,\, \tilde{\boldsymbol{w}}_k)$, where $\{\boldsymbol{\xi}_k\}$ is a stochastic sequence. For example, in the LMS case we have $f(k,\, \tilde{\boldsymbol{w}}_k) = -\boldsymbol{x}_k\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k - \boldsymbol{x}_k v(k)$, and $\boldsymbol{\xi}_k$ would be formed from $\boldsymbol{x}_k$ and $v(k)$. Now define the averaged function f_{av} as

$$f_{av}(k,\, \tilde{\boldsymbol{w}}) = \mathrm{E}\, f(k,\, \tilde{\boldsymbol{w}}),$$

where $\tilde{\boldsymbol{w}}$ is considered *constant* for the computation of the expected value. For example, if $\{\boldsymbol{x}_k\}$ is a stationary sequence, the averaged function for LMS is $f_{av}(k,\, \tilde{\boldsymbol{w}}) = -R\tilde{\boldsymbol{w}}$. Define also the *averaged system*

$$\tilde{\boldsymbol{w}}_{k+1}^{av} = \tilde{\boldsymbol{w}}_k^{av} + \mu f_{av}\big(k,\, \tilde{\boldsymbol{w}}_k^{av}\big)\ , \quad \tilde{\boldsymbol{w}}_0^{av} = \tilde{\boldsymbol{w}}_0\ . \tag{3.8}$$

The fully averaged system does not allow us to predict the steady-state performance of the adaptive algorithm. For this purpose, it is necessary to study the

partially averaged system below,

$$\tilde{\boldsymbol{w}}_{k+1}^{pav} = [I + \mu \nabla_{\tilde{\boldsymbol{w}}} f_{av}(\boldsymbol{0})]\, \tilde{\boldsymbol{w}}_k^{pav} + \mu\big(f(k,\, \boldsymbol{0}) - f_{av}(k,\, \boldsymbol{0})\big)\ , \tag{3.9}$$

where $\nabla_{\tilde{\boldsymbol{w}}} f_{av}(\boldsymbol{0})$ denotes the value of the gradient of f_{av} (with respect to $\tilde{\boldsymbol{w}}$) at the origin. Using the LMS algorithm as an example again, we have

$$\nabla_{\tilde{\boldsymbol{w}}} f_{av}(\boldsymbol{0}) = -R, \qquad \big(f(k,\, \boldsymbol{0}) - f_{av}(k,\, \boldsymbol{0})\big) = -\boldsymbol{x}_k v(k).$$

The following result, proven in [SK95, Ch. 9], shows that if the step-size μ is sufficiently small, the original estimates $\tilde{\boldsymbol{w}}_k$ will remain close to the partially averaged estimates $\tilde{\boldsymbol{w}}_k^{pav}$, and that the steady-state covariance of $\tilde{\boldsymbol{w}}_k$ will be close to that of $\tilde{\boldsymbol{w}}_k^{pav}$.

Theorem 3.4 (Averaging result). *Consider the error equation (3.7) and its averaged forms (3.8) and (3.9), where the sequence $\{\boldsymbol{\xi}_k\}$ is uniform mixing. Assume that (i) the origin, $\boldsymbol{0}$, is an exponentially-stable equilibrium point of the averaged system* (3.8) *with decay rate $O(\mu)$, (ii) the gradient $\nabla_{\tilde{\boldsymbol{w}}} f_{av}(k,\, \tilde{\boldsymbol{w}})$ exists and is continuous at the origin, (iii) there exist constant c and c_1 such that, for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, the following Lipschitz condition holds*

$$\|\nabla_{\tilde{\boldsymbol{w}}} f(k,\, \boldsymbol{a}) - \nabla_{\tilde{\boldsymbol{w}}} f(k,\, \boldsymbol{b})\| \leq c\big\|\boldsymbol{a} - \boldsymbol{b}\big\|.$$

Under these conditions, $\tilde{\boldsymbol{w}}_k$ obtained from (3.7) *satisfies*

$$\limsup_{\mu \to 0} \sup_{k \geq 0}\ \mathrm{E}\big\|\tilde{\boldsymbol{w}}_k - \tilde{\boldsymbol{w}}_k^{pav}\big\| = 0, \quad \textit{for every } \epsilon > 0, \tag{3.10}$$

$$\lim_{\mu \to 0} \lim_{k \to \infty}\ \mathrm{E}\big\|\tilde{\boldsymbol{w}}_k - \tilde{\boldsymbol{w}}_k^{pav}\big\|^2 = 0, \tag{3.11}$$

and

$$\lim_{\mu \to 0} \lim_{k \to \infty} \left(\frac{1}{\mu}\, \mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T\right) = \lim_{\mu \to 0} \lim_{k \to \infty} \left(\frac{1}{\mu}\, \mathrm{E}\, \tilde{\boldsymbol{w}}_k^{pav} \tilde{\boldsymbol{w}}_k^{pav,\, T}\right)\ . \tag{3.12}$$

$\diamondsuit$

Note that this result does not apply to the leaky LMS algorithm, since the origin $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l = \mathbf{0}$ of the averaged leaky LMS error equation is not an exponentially stable equilibrium point (as shown in Sec. 2.3).

3.3.2 Averaging Analysis of LMS

In the special case of the LMS algorithm, the above results can be extended — in particular, it is possible to compute more exact values for the MSE, as in Theorem 3.2. The analysis presented here basically follows from a simple (deterministic) property of LMS. Compute $\|\tilde{\boldsymbol{w}}_k\|^2$:

$$\|\tilde{\boldsymbol{w}}_{k+1}\|^2 = \|\tilde{\boldsymbol{w}}_k\|^2 - 2\mu\big(\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k\big)^2 + \mu^2\|\boldsymbol{x}_k\|^2\big(\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k\big)^2 - \\ - 2\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\boldsymbol{x}_k v(k) + \mu^2\|\boldsymbol{x}_k\|^2 v^2(k).$$

Using Assumption B-1, and adding the above recursion from time $k = 0$ to $k = N$, the following inequality is obtained.

$$\|\tilde{\boldsymbol{w}}_{N+1}\|^2 + \mu(2 - \mu B_x)\sum_{k=0}^{N} e^2(k) \le -2\mu\sum_{k=0}^{N}\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\boldsymbol{x}_k v(k) + \\ + \mu^2\sum_{k=0}^{N}\|\boldsymbol{x}_k\|^2 v^2(k).$$

The averaging argument needs a bounded input sequence to keep the term $2-\mu B_x$ positive. Taking expectations on both sides, the $O(\mu)$ part of the cross-term is zero (by the orthogonality principle), and the analysis is simplified.

The next theorem provides bounds for the performance of LMS. The proof in [Sol89] uses the above relation and the assumption:

B-2. *The sequence* $\{\boldsymbol{x}_k\}$ *is purely nondeterministic.*

If $\boldsymbol{x}_k$ is formed from a tap-delay line (as in NDS-2), then this assumption means that the newest element of $\boldsymbol{x}_k$, $a(k)$, cannot be perfectly predicted from the previous elements $a(k-i)$. In other words, if $\hat{a}(k)$ is any estimator of $a(k)$ given $\{a(k-1) \ \ldots \ a(0)\}$, then there is a constant $\sigma^2 > 0$ such that

$$\operatorname{var}\bigl(\hat{a}(k) - a(k)\bigr) \geq \sigma^2 > 0.$$

The theorem also assumes that the input sequence is stationary, i.e.,

IS-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

With these assumptions, the following results hold (see also [Ben87]).

Theorem 3.5. *Under Assumptions B-1–B-2 and IS-1,*

$$\mathrm{E}\, e(\infty)^2 = \sigma_v^2 + \frac{1}{2}\mu m f \sigma_v^2 \operatorname{Tr} R + O(\mu^2),$$

where

$$m = 1 - \frac{2}{\mathrm{E}\bigl(v^2(0)\|\boldsymbol{x}_0\|^2\bigr)} \sum_{k=1}^{\infty} \mathrm{E}\bigl(v(k)v(0)\boldsymbol{x}_k^T \Phi_{k,0} \boldsymbol{x}_0\bigr),$$
$$f = \frac{\mathrm{E}\bigl(v^2(0)\|\boldsymbol{x}_0\|^2\bigr)}{\sigma_v^2 \operatorname{Tr} R},$$

and $\Phi_{k,0} = \prod_{i=0}^{k}\bigl(I - \mu \boldsymbol{x}_i \boldsymbol{x}_i^T\bigr)$ *is the state transition matrix of the linear time-varying error equation* (2.1).

The constant m *can be expanded as*

$$m = 1 - 2\frac{\sum_{k=1}^{\infty} \mathrm{E}\bigl(v(k)v(0)\boldsymbol{x}_k^T \boldsymbol{x}_0\bigr)}{\mathrm{E}\bigl(v^2(0)\|\boldsymbol{x}_0\|^2\bigr)} + O(\mu).$$

♢

These results are essentially of the same nature as in Secs. 3.1 and 3.2. The only difference is in the assumptions — now the N-dependent assumption NDS-2

is relaxed to allow for the more general class of uniform-mixing processes, but with the new requirement imposed by the bounded input assumption B-2.

Similar results exist for the case of time-variant $\{\boldsymbol{w}_{*,k}\}$ (but still with stationary $\boldsymbol{x}_k$) [Sol89]. Unfortunately some quantities are left without definition in [Sol89], so we do not list the result here.

The case of time-variant $\boldsymbol{x}_k$ is studied in [BA80a, BA80b, Bit83, Bit81, CG91, Guo94] and [Sol94a]. In these works, the input sequence is $\boldsymbol{x}_k$ is assumed to be time-variant and uniformly bounded (i.e., Assumption B-1 holds). These works are very technical, and only confirm the results presented in the previous sections for a slightly more general class of inputs; for this reason we do not discuss them further.

3.4 PERFORMANCE AND CONVERGENCE OF NLMS

Several of the above results can be extended to the normalized-LMS algorithm, if the model of Section 2.4 is used. The averaging analysis is particularly adequate for the study of the NLMS algorithm, given that the assumption of bounded inputs is always verified in NLMS (recall that using the change of variables of Sec. 2.4, the input sequence for NLMS is $\left\{\boldsymbol{x}_k/\sqrt{a+\|\boldsymbol{x}_k\|^2}\right\}$). Therefore, the averaging theory results for the NLMS algorithm are more general than the results following Theorems 3.2 and 3.3: these theorems require an N-dependent input sequence, while averaging requires only bounded inputs (an assumption that is always satisfied) and the less restrictive uniform-mixing condition.

The transformed recursion for the normalized LMS algorithm (which we re-

peat below) satisfies all the conditions for Thms. 3.3, 3.4, and 3.5:

$$\boldsymbol{w}_{k+1}^n = \boldsymbol{w}_k^n + \frac{\mu}{a + \|\boldsymbol{x}_k\|^2}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k^n\big).$$

Recall that the transformed variables are

$$\boldsymbol{x}_k^{(n)} \triangleq \frac{\boldsymbol{x}_k}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}, \quad v^{(n)}(k) \triangleq \frac{v(k)}{\sqrt{a + \|\boldsymbol{x}_k\|^2}}, \quad \text{and} \quad y^{(n)}(k) \triangleq \frac{y(k)}{\sqrt{a + \|\boldsymbol{x}_k\|^2}},$$

and the transformed covariance matrix is

$$R^{(n)} \triangleq \mathrm{E}\, \boldsymbol{x}_k^{(n)} \boldsymbol{x}_k^{(n)\,T}.$$

Theorem 3.6. *Under Assumptions I-3, UM-1, B-1–B-2, and IS-1, the normalized LMS algorithm with $a > 0$ is mean-square stable, and the steady-state MSD satisfies, when $\mu \approx 0$,*

$$\mathcal{D} \approx \frac{\mu \sigma_v^2}{2} \operatorname{Tr}\left\{ R^{(n)^{-1}} \mathrm{E}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\left(a + \|\boldsymbol{x}_k\|^2\right)^2} \right) \right\}, \tag{3.13}$$

In addition, the following equality also holds:

$$\lim_{k \to \infty} \mathrm{E}\, e^{(n)}(k)^2 = \mathrm{E}\, \frac{\sigma_v^2}{a + \|\boldsymbol{x}_k\|^2} + \frac{1}{2} \mu {\sigma_v}^2 \operatorname{Tr}\left\{ \mathrm{E}\left(\frac{\boldsymbol{x}_k \boldsymbol{x}_k^T}{\left(a + \|\boldsymbol{x}_k\|^2\right)^2} \right) \right\} + O(\mu^2), \tag{3.14}$$

Proof: The MSD is obtained by direct application of Thm. 3.4 to the NLMS algorithm. It is clear from the NLMS recursion that f and f_{av} are differentiable functions when $a > 0$. Therefore, Thm. 3.4 holds, and the MSD is, up to first order in μ, equal to the value given in (3.13).

Note that (3.14) is not the MSE, but the expectation of a normalized version of $e(k)^2$. In general, there is no simple relation between the two, unless $\{\boldsymbol{x}_k\}$ is iid or $\|\boldsymbol{x}_k\|$ is constant. Unfortunately, the expression for $\mathrm{E}\, e(k)^2$ cannot be computed simply by using our change of variables (the argument for the computation of $\mathrm{E}\, e(k)^2$ in the proof of Theorem 3.5 does not extend to this case). We are not

aware of a result in the literature extending Thm. 3.5 to the normalized LMS algorithm with $\mathrm{E}\, e(k)^2$ instead of $\mathrm{E}\, e^{(n)}(k)^2$.

$\diamondsuit$

3.5 LEAKY LMS ALGORITHM

The extension of the results of Secs. 3.1–3.3 to the leaky-LMS algorithm is not a simple task, basically because the constant term $\mu\alpha\boldsymbol{w}_*$ in the leaky LMS error equation (2.39) eliminates the exponentially-stable equilibrium point at $\mathbf{0}$ that existed in the LMS error equation (2.1). This directly violates one of the main assumptions of Thm. 3.4 (see the quote from [Set93] in Sec. 1.4). We circumvent this difficulty by developing a new leaky algorithm, which we describe in Chapter 6. In fact, this new algorithm not only makes the analysis possible, but it actually has a better performance than leaky LMS. See Chapter 6, Table 6.1, and Figs. 6.4, 6.5, and 6.6.

3.6 CONTRIBUTIONS OF THIS CHAPTER

The extension of Theorem 3.5 to the NLMS algorithm using the change of variables described in Sec. 2.4 is new. A stability analysis of the NLMS algorithm without independence assumptions was given for non-stationary $\{\boldsymbol{x}_k\}$ in [Guo94]. Nevertheless, unlike this reference, our analysis provides computable performance estimates.

As we mentioned in Sec. 1.4, the extension of the results described in this chapter to the leaky LMS algorithm is a difficult open problem. Nevertheless,

we believe that this problem has been rendered virtually meaningless, since we develop a new algorithm that in many ways surpasses the performance of leaky LMS in Chapter 6 [NS99c]. For this new algorithm, we extend the results of Sec. 3.3, see Sec. 6.6.

APPENDIX FOR CHAPTER 3

3.A TIME-VARIANT N-DEPENDENT INPUTS

We provide here the extension of Theorem 3.3 to the non-stationary case. As we did in Appendix 2.D, it is also necessary to describe the time variation of $\tilde{\boldsymbol{w}}_{*,k}$ and of the input sequence $\{\boldsymbol{x}_k\}$. The first assumption below substitutes R-1, saying that the smallest eigenvalue of the covariance matrix of $\boldsymbol{x}_k$ is *uniformly* lower bounded by a positive quantity.

NDV-1. *There exists a positive constant λ_m such that, for all k,*

$$\lambda_{\min}\Big(\mathrm{E}\big(\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\Big) \geq \lambda_m.$$

The next assumption limits the speed of variation of $\boldsymbol{w}_{*,k}$, saying that

$$\|\boldsymbol{w}_{*,k+1} - \boldsymbol{w}_{*,k}\|$$

is *deterministically* upper bounded by a quantity Δ. This way of limiting the speed of the time variation is different than what was done in Appendix 2.D by means of a parameter θ. Accordingly, the bound for the MSE obtained here will be different.

NDV-2. *The difference*

$$\boldsymbol{\Delta}_k \stackrel{\Delta}{=} \boldsymbol{w}_{*,k+1} - \boldsymbol{w}_{*,k} \tag{3.A.1}$$

satisfies $\sup_{k\geq 0}\|\boldsymbol{\Delta}_k\| = \Delta < \infty$.

The proof of the next theorem also requires that several moments of the input sequence be bounded. The assumption below replaces NDS-4–NDS-5 for the non-stationary case.

NDV-3. *There exist positive numbers K_1, K_2 such that*

$$\begin{aligned} \mathrm{E}\Big(\big\|\boldsymbol{x}_k\big\|^{24N}\Big) &\leq K_1 \\ \mathrm{E}\Big(\big|y(k)\big|^{4+\frac{4}{6N-1}}\Big) &\leq K_2 \end{aligned} \qquad \forall\, k \geq 0.$$

Theorem 3.A.1. *If Assumptions NDV-1–NDV-3 hold, then there exist positive numbers $\big(\mu_0,\, \beta,\, \beta^{'}\big)$ such that $\tilde{\boldsymbol{w}}_k$ given by the LMS algorithm* (1.6) *satisfies*

$$\limsup_{k\to\infty} \mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2 \leq \beta\mu + \beta^{'}\frac{\Delta^2}{\mu^2}, \quad \forall\, \mu \in (0, \mu_0).$$

$\diamondsuit$

The proof in [EM85] does not give estimates for the parameters β and $\beta^{'}$. This means that one does not obtain any actual performance information from this result, besides learning that the misadjustment does not go to zero as the step-size is decreased (compare with the results obtained in Appendix. 2.D). The difference between the two results here and in Appendix. 2.D is due to the different models of parameter variation adopted. In this section, the parameter varies according to a deterministic model, while in Appendix. 2.D a stochastic model was used. Note also that the model used in Theorem 3.A.1 allows for a time-variant input sequence $\big\{\boldsymbol{x}_k\big\}$, while in Appendix 2.D the time-variation was due to $\boldsymbol{w}_{*,\,k}$ only.

CHAPTER 4

MEAN-SQUARE STABILITY OF LMS WITHOUT SLOW ADAPTATION APPROXIMATIONS

The results presented so far give a good understanding of the behavior of the LMS algorithm when the step-size μ is sufficiently small (but without quantifying how small μ should be). However, from the recursion (1.6), an infinitesimal step-size ($\mu \approx 0$) implies that the weight estimates $\boldsymbol{w}_k$ change very slowly at each iteration, and consequently the convergence rate is small. This is inconvenient, since in many applications the filter output cannot be used before the MSE $\mathrm{E}\, e(k)^2$ achieves a sufficiently low level. Slow adaptation is also annoying in non-stationary environments, where a slow convergence rate may not allow the algorithm to properly track time variations in signal statistics. A designer might then wish to employ larger step-sizes to improve the convergence speed of the algorithm, especially during the initial convergence phase (before steady-state is achieved).

The following questions are therefore relevant and remain largely unanswered in the literature.

(i) How small must the step-size be so that the independence-based approximations are still reasonable ? Also, for a given value of the step-size, what

is the order of magnitude of the error incurred by using these approximations ?

(ii) What is the real performance of the adaptive algorithm when the step-size is not small ?

(iii) How large the step-size can get without compromising filter (mean-square) stability ?

(iv) What is the step-size that gives the fastest convergence rate ?

For step-sizes that are not infinitesimally small, there are very few results in the literature that predict or confirm the behavior/stability of the LMS algorithm and its variants (see, e.g., the statement from [Slo93] in Sec. 1.4 regarding this issue).

The purpose of this chapter is to develop a new method to study the stability of the LMS algorithm, without relying on the independence assumptions and without assuming beforehand that the step-size is vanishingly small. Since an exact expression for the largest step-size (say, $\mu_{\max}$) that addresses point (iii) above is difficult to obtain, we shall instead derive an upper bound $\bar{\mu}$ on how large the step-size can be for mean-square stability (say, $\mu < \bar{\mu} \leq \mu_{\max}$). While our bound $\bar{\mu}$ is not tight (i.e., close to $\mu_{\max}$) at this stage of our analysis, it is, to the authors' knowledge, the first computable bound to be obtained. It is applicable to a generic distribution of the input sequence (and in particular, it allows for a normally distributed input).

The significance of this work is therefore in developing a framework that studies filter stability without resorting to slow adaptation approximations, and without assuming that the input sequence is iid. The major contribution is the fact that our bound is computable, i.e., our result is not only an existence result.

Given an input distribution and a filter length, a bound $\bar{\mu}$ can be easily computed via a maximization routine (see Sec. 4.4).

Our discussion builds upon an approach originally suggested in [FF86], and which will lead naturally to a state-space framework. Basically, the arguments we employ in the future sections can be summarized as follows. We first find a dynamic state-space model for the evolution of the covariance matrix $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$; the states of this model will consist of the entries of the covariance matrix in addition to several other relevant quantities. The state equation will be of the form

$$\boldsymbol{\Gamma}_{k+1} = \Phi(\mu)\boldsymbol{\Gamma}_k + \boldsymbol{b}, \tag{4.1}$$

where $\boldsymbol{b}$ is a constant vector, $\boldsymbol{\Gamma}_k$ is the state vector, and Φ is a constant matrix. With this model, the largest step-size ($\mu_{\max}$) that guarantees stable performance of the LMS filter (and therefore answers point (iii) above) will be the largest μ for which $\Phi(\mu)$ is still a stable matrix, i.e.,

$$\mu_{\max} \stackrel{\Delta}{=} \sup\{\mu \ \text{ such that } \rho\big(\Phi(\mu)\big) < 1\}\ , \tag{4.2}$$

where $\rho(\Phi)$ denotes the *spectral radius* of Φ, i.e., $\rho(\Phi) = \max_i \big|\lambda_i\big(\Phi\big)\big|$.

Unfortunately, determining $\mu_{\max}$ in (4.2) is not a trivial task for two main reasons. First, the eigenvalues of the matrix Φ depend nonlinearly on the step size μ and, secondly, the dimension of Φ grows extremely fast with the filter length (for example, for $M = 6$ the matrix has size $28,181 \times 28,181$). It is therefore computationally infeasible to work directly with Φ; the approach is feasible only for relatively small filter lengths. For this reason, reference [FF86] considered only the case $M = 2$ (i.e., a filter with two taps), while reference [DP95] used the same method for orders up to $M = 6$ coupled with a numerical procedure (viz., the power method for sparse matrices) for the evaluation of the eigenvalues of

Φ. For larger filter lengths, we need to develop an alternative procedure for the estimation of $\mu_{\max}$ that does not work directly with the matrix Φ.

The approach we propose in this chapter is based on the observation that the matrix Φ, although of large dimensions, is both *sparse* and *structured*. These two properties combined can be used to derive a bound on the step-size for stable performance. [Moreover, the bound will be such that it is not a function of the maximum value that $\|\boldsymbol{x}_k\|$ can attain; the result will depend only on the distribution of the input sequence, and on average quantities.]

We start our discussion by studying in some detail the simple case $M = 2$ (i.e., a filter with 2 taps). This will allow us to explain in a fairly clear manner most of the steps involved in our construction. In a latter section (Sec. 4.2) we shall extend the discussion to larger values of M. It will be seen then that the dimensions of the matrices we work with grow exponentially fast with M, so that we shall not be able to show all the details in explicit form. Instead, we shall establish several structural and sparsity properties and exploit them. Moreover, some new phenomena arise for larger values of M that do not occur for the simple case $M = 2$. We shall also address these issues.

4.1 THE STATE-SPACE MODEL FOR $M = 2$

We assume that $\{y(k), \boldsymbol{x}_k\}$ are related via a linear model of the form (1.11), with a zero-mean iid noise sequence $v(k)$ that is independent of the input sequence. Using the LMS update (1.6) and the model (1.11), we find that the error equation for LMS is given by

$$\tilde{\boldsymbol{w}}_{k+1} = (I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T)\tilde{\boldsymbol{w}}_k - \mu \boldsymbol{x}_k v(k) \ . \tag{4.3}$$

We also suppose that the regressors $\{\boldsymbol{x}_k\}$ arise from a tap-delay line, say

$$\boldsymbol{x}_k = \begin{bmatrix} a(k-1) & a(k) \end{bmatrix}^T. \tag{4.4}$$

where the input sequence $\{a(k)\}$ is assumed iid, with zero mean, and moments $\sigma_p = \mathrm{E}\, a(k)^p$, for $p \geq 1$ (note that zero-mean $a(k)$ means $\sigma_1 = 0$). These assumptions imply that

$$\boxed{\tilde{\boldsymbol{w}}_k \text{ is independent of } a(k)} \tag{4.5}$$

but not of $a(k-1)$. This is the basic property that will yield a linear state-space model.

Note: for simplicity, in this section only we shall assume that $\sigma_p = 0$ for all odd values of p, i.e., $0 = \sigma_3 = \sigma_5 = \ldots$.

We are interested in conditions under which the MSD is bounded (i.e., conditions under which $\mathrm{Tr}\big(\mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T\big)$ forms a bounded sequence). This requires that we study the stability of the matrix Φ in (4.1) and determine conditions under which its eigenvalues are strictly inside the unit disc. It turns out that, under the above assumptions, the noise sequence $\{v(k)\}$ does not influence the stability of the recursion (4.1) since it only affects the driving term $\boldsymbol{b}$. For this reason, in this chapter we shall assume $v(k) \equiv 0$.

4.1.1 Obtaining the Linear Model

We present briefly the main steps of the derivation of the linear model in an $M = 2$ example [FF86] in order to highlight the main steps. Later we consider the case of a general M. Let $\tilde{w}_{k,i}$ represent the i-th entry of $\tilde{\boldsymbol{w}}_k$, that is,

$$\tilde{\boldsymbol{w}}_k = \begin{bmatrix} \tilde{w}_{k,1} & \tilde{w}_{k,2} \end{bmatrix}^T.$$

We want to find a state-space model describing the evolution of the entries of $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$, which we denote by

$$\gamma_1(k) \triangleq \mathrm{E}\,\tilde{w}_{k,1}^2, \qquad \gamma_2(k) \triangleq \mathrm{E}\,\tilde{w}_{k,2}^2, \qquad \gamma_3(k) \triangleq \mathrm{E}\Big(\tilde{w}_{k,1}\tilde{w}_{k,2}\Big).$$

That is,

$$\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T = \begin{bmatrix} \gamma_1(k) & \gamma_2(k) \\ \gamma_2(k) & \gamma_3(k) \end{bmatrix}.$$

These variables will be some of the elements of our state vector $\boldsymbol{\Gamma}_k$.

Using the LMS error equation (4.3) to compute $\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$, we obtain

$$\begin{aligned} \tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T = \tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T - \mu\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\boldsymbol{x}_k\boldsymbol{x}_k^T + \\ + \mu^2\boldsymbol{x}_k\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\boldsymbol{x}_k\boldsymbol{x}_k^T. \end{aligned} \tag{4.6}$$

The $(1,1)$ element of the above matrices provides an expression for $\tilde{w}_{k+1,1}^2$ as a function of quantitites available at time k:

$$\begin{aligned} \tilde{w}_{k+1,1}^2 = \big(1-\mu a(k-1)^2\big)^2\tilde{w}_{k,1}^2 - 2\mu a(k-1)a(k)\big(1-\mu a(k-1)^2\big)\tilde{w}_{k,1}\tilde{w}_{k,2} + \\ + \mu^2 a(k-1)^2a(k)^2\tilde{w}_{k,2}^2. \end{aligned} \tag{4.7}$$

Taking expectations and using the independence of $a(k)$ from $\big\{a(k-1),\ \tilde{\boldsymbol{w}}_k\big\}$, it follows that

$$\begin{aligned} \gamma_1(k+1) = \gamma_1(k) - 2\mu\,\mathrm{E}\big(a(k-1)^2\tilde{w}_{k,1}^2\big) + \\ + \mu^2\,\mathrm{E}\big(a(k-1)^4\tilde{w}_{k,1}^2\big) + \mu^2\sigma_2\,\mathrm{E}\big(a(k-1)^2\tilde{w}_{k,2}^2\big). \end{aligned}$$

The right-hand side of this recursion does not depend only on $\gamma_1(k)$ to $\gamma_3(k)$, but also on

$$\begin{aligned} &\mathrm{E}\big(a(k-1)^2\tilde{w}_{k,1}^2\big), && && \mathrm{E}\big(a(k-1)^4\tilde{w}_{k,1}^2\big), \\ &\mathrm{E}\big(a(k-1)^2\tilde{w}_{k,2}^2\big), && \text{and} && \mathrm{E}\big(a(k-1)^2\tilde{w}_{k,1}\tilde{w}_{k,2}\big). \end{aligned}$$

To proceed, we need to find recursions for these quantities. Note that $a(k-1)$ is the first element of the vector $\boldsymbol{x}_k$, that is, $x_{k,1} = a(k-1)$. The idea is to use this relation to define the new variables

$$\gamma_4(k) \triangleq \mathrm{E}\left(x_{k,1}^2 \tilde{w}_{k,1}^2\right), \qquad \gamma_5(k) \triangleq \mathrm{E}\left(x_{k,1}^4 \tilde{w}_{k,1}^2\right),$$
$$\gamma_6(k) \triangleq \mathrm{E}\left(x_{k,1}^2 \tilde{w}_{k,2}^2\right), \qquad \gamma_7(k) \triangleq \mathrm{E}\left(x_{k,1}^2 \tilde{w}_{k,1} \tilde{w}_{k,2}\right),$$

and to obtain expressions for these variables at time $k+1$. For example,

$$\gamma_4(k+1) = \mathrm{E}\left(x_{k+1,1}^2 \tilde{w}_{k+1,1}^2\right).$$

An expression for $\gamma_4(k+1)$ can be obtained by multiplying both sides of (4.7) by $x_{k+1,1}^2 = x_{k,2}^2 = a(k)^2$. Using (4.5) to compute the expectation, we get

$$\gamma_4(k+1) = \sigma_2 \gamma_1(k) - 2\mu\sigma_2\gamma_4(k) + \sigma_2\gamma_5(k) + \mu^2\sigma_4\gamma_6(k).$$

After finding similar recursions for all variables, we obtain the linear model

$$\boldsymbol{\Gamma}_{k+1} = \Phi(\mu)\boldsymbol{\Gamma}_k, \tag{4.8}$$

where

$$\boldsymbol{\Gamma}_k = \begin{bmatrix} \gamma_1(k) & \gamma_2(k) & \ldots & \gamma_7(k) \end{bmatrix}^T, \tag{4.9}$$

and

$$\Phi = \begin{bmatrix} 1 & -2\mu & 0 & 0 & \mu^2 & \mu^2\sigma_2 & 0 \\ \sigma_2 & -2\mu\sigma_2 & 0 & 0 & \mu^2\sigma_2 & \mu^2\sigma_4 & 0 \\ 0 & \mu^2\sigma_2 & 1-2\mu\sigma_2+\mu^2\sigma_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-\mu\sigma_2 & 0 & 0 & -\mu\big(1-2\mu\sigma_2\big) \\ \sigma_4 & -2\mu\sigma_4 & 0 & 0 & \mu^2\sigma_4 & \mu^2\sigma_6 & 0 \\ 0 & \mu^2\sigma_4 & \sigma_2-2\mu\sigma_4+\mu^2\sigma_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_2-\mu\sigma_4 & 0 & 0 & -\mu\big(\sigma_2-2\mu\sigma_4\big) \end{bmatrix}. \tag{4.10}$$

Note that even for $M = 2$, the state-space model has already 7 states, and the matrix Φ is fairly sparse.

4.1.2 Mean-Square Stability Analysis

Having formed Φ in (4.10), the mean-square stability of the adaptive filter can now be completely characterized by studying how the spectrum of Φ varies with the step-size μ. In principle, we can proceed to determine the largest μ, say $\mu_{\max}$, for which all the eigenvalues of Φ will be strictly inside the unit circle. Although this can be done easily for $M = 2$, we shall see in later sections that the size of Φ grows exponentially fast with M (see Table 4.1). This creates two main difficulties:

1. The computation of the eigenvalues of Φ for each μ becomes computationally demanding.

2. The eigenvalues of Φ depend very nonlinearly on μ.

For this reason, we propose in this work an alternative procedure that avoids working with the eigenvalues of Φ altoghether. In particular, we shall show that, for any M, the computation of the norm $\|\Phi\|_\infty$ (the maximum row sum of Φ) can be performed in only $O(M)$ operations, due to the structure of Φ. We then use this fact to compute a bound $\bar{\mu}$ to $\mu_{\max}$. The idea is to use the inequality (see [HJ87] and Sec. 4.4 for more details)

$$\rho(\Phi) \stackrel{\Delta}{=} \max_i \left|\lambda_i\big(\Phi\big)\right| \leq \|\Phi\|_\infty, \tag{4.11}$$

where $\rho(\Phi)$ is the *spectral radius* of Φ, and $\lambda_i\big(\Phi\big)$ is the i-th eigenvalue of Φ. That is, the maximum eigenvalue of Φ in magnitude is always bounded by $\|\Phi\|_\infty$. Therefore, a sufficient condition to guarantee that all λ_i are strictly less than one is to find the range of μ for which $\|\Phi(\mu)\|_\infty < 1$.

Unfortunately, directly applying this idea to Φ yields no useful results. To understand this, simply refer again to (4.10). Note that the absolute sum of the first row of (4.10) is

$$\|\Phi\|_\infty \geq \sum_{i=1}^{7} \left| \left(\Phi\right)_{1,i} \right| = 1 + 2\mu + \mu^2 + \mu^2\sigma_2 > 1 \quad \text{for all } \mu > 0.$$

That is, $\|\Phi\|_\infty$ is always larger than unity regardless of μ! This is because some rows of Φ have unit entries (this property will in fact hold for *all* M, not just $M = 2$). Therefore, there is no value of μ for which $\|\Phi\|_\infty$.

Recall, however, that similarity transformations do not alter the spectral radius of a matrix, i.e.,

$$\rho(\Phi) = \rho\left(T^{-1}\Phi T\right) \quad \text{for any invertible } T.$$

Nevertheless, they do alter the ∞-norm of a matrix. Our second idea is therefore to show how to construct a similarity transformation T such that there would exist a $\bar{\mu}$ such that

$$\left\|T^{-1}\Phi(\mu)T\right\|_\infty < 1$$

for all $\mu < \bar{\mu}$.

We shall illustrate this procedure for the $M = 2$ case below, before considering the general case in Sec. 4.4. We first need to define the row and column operators below. Define the square matrix

$$[I]^{\alpha}_{i,j} \triangleq \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & & \ddots & & & & & & \vdots \\ 0 & 0 & 0 & \dots & 1 & \dots & \alpha & \dots & 0 \\ \vdots & & & & & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{bmatrix} \begin{matrix} \\ \\ \\ \xleftarrow{\text{row } i} \\ \\ \\ \end{matrix}$$

$$\uparrow \text{ column } j$$

where the entry α appears in the off-diagonal position (i, j) (i.e., $i \neq j$). This matrix is a simple modification of the identity, with the (i, j)-th off-diagonal 0 replaced by α. The effect of pre-multiplying a square matrix A by $[I]^{\alpha}_{i,j}$, i.e.,

$$[I]^{\alpha}_{i,j} A,$$

is to adding α times the j-th row of A to the i-th row of A. Similarly, post-multiplying A by $[I]^{\alpha}_{i,j}$ corresponds to adding α times the i-th column of A to its j-th column (note that the indices i and j are interchanged in the row and column operation — in the row operation, it is the i-th row that is modified, whereas the column operation modifies the j-th column of A).

In addition, we note that

$$\left([I]^{\alpha}_{i,j}\right)^{-1} = [I]^{-\alpha}_{i,j}.$$

Therefore, the combination of row and column operations

$$[I]^{\alpha}_{i,j} \; A \; [I]^{-\alpha}_{i,j}$$

is a similarity transformation, and it does not change the eigenvalues of A. It only modifies row i and column j of A.

We can return now to transforming Φ in (4.10). Note that the second row of (4.10) is very similar to its first row, in the sense that the element $(\Phi)_{1,i}$ on the first row is nonzero only if the element $(\Phi)_{2,i}$ on the second row is nonzero. In addition, the second row can be obtained from the first by inspection — to move from row 1 to row 2, we simply need to replace σ_p by σ_{p+2} (elements that are not multiples of any σ_p in row 1, such as -2μ, should be treated as if multiplied by $\sigma_0 \stackrel{\Delta}{=} 1$).

Therefore, if we apply the row operation $[I]_{1,2}^{-2\mu}$ to replace the entry 1 in the position $(1,1)$ in (4.10) with $1-2\mu\sigma_2$, and complete the similarity transformation with the column operation $[I]_{1,2}^{2\mu}$, we obtain

$$\Phi^{(1)} = \begin{bmatrix} 1-2\mu\sigma_2 & 0 & 0 & 0 & \mu^2(1-2\mu\sigma_2) & \mu^2(\sigma_2-2\mu\sigma_4) & 0 \\ \sigma_2 & 0 & 0 & 0 & \mu^2\sigma_2 & \mu^2\sigma_4 & 0 \\ 0 & \mu^2\sigma_2 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-\mu\sigma_2 & 0 & 0 & -\mu(1-2\mu\sigma_2) \\ \sigma_4 & 0 & 0 & 0 & \mu^2\sigma_4 & \mu^2\sigma_6 & 0 \\ 0 & \mu^2\sigma_4 & a_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_2-\mu\sigma_4 & 0 & 0 & -\mu(\sigma_2-2\mu\sigma_4) \end{bmatrix},$$

where we have defined

$$\Phi^{(1)} \stackrel{\Delta}{=} [I]_{1,2}^{-2\mu}\Phi[I]_{1,2}^{2\mu}, \qquad a_1 \stackrel{\Delta}{=} 1-2\mu\sigma_2+\mu^2\sigma_4, \qquad a_2 \stackrel{\Delta}{=} \sigma_2-2\mu\sigma_4+\mu^2\sigma_6.$$

Note that while the row operation replaced the 1 entry in the $(1,1)$ position of Φ with $1-2\mu\sigma_2$, the column operation replaced the -2μ entry in the $(1,2)$ position with 0. Both changes are important, since now, except for the $(1,1)$ entry, all terms in the first row of $\Phi^{(1)}$ are $O(\mu^2)$, reducing the absolute sum of the first row of the transformed matrix to

$$\sum_{i=1}^{7}\left|\left(\Phi^{(1)}\right)_{1,i}\right| = |1-2\mu\sigma_2|+\mu^2|1-2\mu\sigma_2|+\mu^2|\sigma_2-2\mu\sigma_4| < 1 \qquad (4.12)$$

for sufficiently small μ.

It is important to note that the above transformation modified only 3 rows of Φ: besides rows 1 and 2, only row 5 was modified. In addition, row 5 can also be obtained from row 1 by inspection, this time replacing σ_p in row 1 with σ_{p+4} in row 5. This observation will carry over to general filter lengths $M > 2$ — in general, several similarity transformations will be necessary, but since only a few, very specific rows are modified by each transformation, we are able to keep track of all the effects of the transformations without ever having to form the matrix Φ explicitly.

We still need to check the other rows of $\Phi^{(1)}$. The absolute sum of row 2 is

$$\sum_{i=1}^{7} \left| \left(\Phi^{(1)}\right)_{2,i} \right| = \sigma_2 + \mu^2\sigma_2 + \mu^2\sigma_4. \tag{4.13}$$

Therefore, $\|\Phi^{(1)}\|_\infty < 1$ requires that $\sigma_2 < 1$. This apparently restrictive condition can be dealt with via a simple change of variables (that we describe in full detail in Appendix 4.F, equation (4.F.1)). The idea is to split μ as

$$\mu \triangleq \mu_f \mu_a,$$

and to apply the change of variables

$$\bar{y}(k) \triangleq \sqrt{\mu_a}\, y(k), \qquad \bar{\boldsymbol{x}}_k \triangleq \sqrt{\mu_a}\, \boldsymbol{x}_k \qquad \bar{a}(k) \triangleq \sqrt{\mu_a}\, a(k). \tag{4.14}$$

These tranformations do not modify the LMS recursion, which now reads

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu_f \bar{\boldsymbol{x}}_k \big(\bar{y}(k) - \boldsymbol{w}_k^T \bar{\boldsymbol{x}}_k\big).$$

That is, the estimates $\boldsymbol{w}_k$ are *not* modified by the above transformations. Their effect is only to modify the moments of $a(k)$, since now

$$\bar{\sigma}_p \triangleq \mathrm{E}\,(\bar{a}(k)^p) = \mu_a^{p/2}\sigma_p.$$

Therefore, we can always choose μ_a small enough so that $\bar{\sigma}_p < 1$, allowing us to satisfy condition (4.13).

Note also that the absolute sum of row 5 imposes a similar condition, viz.,

$$\sum_{i=1}^{7} \left| \left(\Phi^{(1)}\right)_{5,i} \right| = \sigma_4 + \mu^2\sigma_4 + \mu^2\sigma_6 < 1.$$

Again, this condition can be satisfied by choosing a small enough μ_a. To keep the discussion as simple as possible, we will defer the application of the change of variables until all similarity transformations have been found. For now, we will just assume that $\sigma_2 < 1$ and $\sigma_4 < 1$.

Let us verify now the absolute sum of row 3. This row (which is the same in Φ and $\Phi^{(1)}$) does not need any modification, since

$$\sum_{i=1}^{7} \left| \left(\Phi^{(1)}\right)_{4,i} \right| = \left|1 - 2\mu\sigma_2 + \mu^2\sigma_4\right| + \mu^2\sigma_2 < 1$$

for small enough μ. We can proceed to the next row.

Computing the absolute sum of row 4 in $\Phi^{(1)}$ (or in Φ, since both matrices have identical fourth rows), we obtain

$$\sum_{i=1}^{7} \left| \left(\Phi^{(1)}\right)_{4,i} \right| = |1 - \mu\sigma_2| + \mu|1 - 2\mu\sigma_2|.$$

Although it is possible to find a range of μ for which the above row sum is strictly less than 1, this will not be the case for similar rows with larger filter lengths. Even in this example, a second similarity transformation can allow for a larger range of μ, as we show next.

Note that the last row (row 7) of $\Phi^{(1)}$ can be obtained from row 4 by inspection (just as row 2 could be obtained by inspection from row 1 in Φ). If we apply the row operation $[I]_{4,7}^{-\mu}$, and (to complete the similarity transformation) the column operation $[I]_{4,7}^{\mu}$, we obtain

$$\Phi^{(2)} \triangleq [I]_{4,7}^{-\mu}\, \Phi^{(1)}\, [I]_{4,7}^{\mu} =$$

$$= \begin{bmatrix} 1-2\mu\sigma_2 & 0 & 0 & 0 & \mu^2(1-2\mu\sigma_2) & \mu^2(\sigma_2-\mu\sigma_4) & 0 \\ \sigma_2 & 0 & 0 & 0 & \mu^2\sigma_2 & \mu^2\sigma_4 & 0 \\ 0 & \mu^2\sigma_2 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_1 & 0 & 0 & \mu^2(\sigma_2-\mu\sigma_4) \\ \sigma_4 & 0 & 0 & 0 & \mu^2\sigma_4 & \mu^2\sigma_6 & 0 \\ 0 & \mu^2\sigma_4 & a_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_2-\mu\sigma_4 & 0 & 0 & \mu^2\sigma_4 \end{bmatrix},$$

where a_1 and a_2 are as in the definition of $\Phi^{(1)}$.

Again, the transformation not only replaced $1-\mu\sigma_2$ with the quantity $1-2\mu\sigma_2+\mu^2\sigma_4$, but also replaced the $O(\mu)$ entry in position $(4,7)$ with an $O(\mu^2)$ entry. Now the absolute sum of the fourth row is

$$\sum_{i=1}^{7} \left| \left(\Phi^{(2)}\right)_{4,i} \right| = |1-2\mu\sigma_2+\mu^2\sigma_4| + \mu^2|\sigma_2-\mu\sigma_4|,$$

which is less than 1 for $\mu < 2\sigma_2/(\sigma_2+\sigma_4)$ (if $1/4 > \sigma_2 > \sigma_4$). Note that the only rows modified by this second transformation were rows 4 and 7.

A last transformation can be applied to $\Phi^{(2)}$. Although this transformation does not achieve significant improvement in this small example, it will be helpful for larger M. The idea is to eliminate as many $O(\mu^2)$ entries in the first row of $\Phi^{(2)}$ as possible, and replace them with $O(\mu^3)$. In this example, we can add μ^2 times the 5th row of $\Phi^{(2)}$ to its first row, and complete the similarity transformation with the column transformation $[I]_{1,5}^{-\mu^2}$ to obtain

$$\Phi^{(3)} \triangleq \begin{bmatrix} a_1 & 0 & 0 & 0 & 0 & \mu^2 a_2 & 0 \\ \sigma_2 & 0 & 0 & 0 & 0 & \mu^2\sigma_4 & 0 \\ 0 & \mu^2\sigma_2 & a_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{a}_1 & 0 & 0 & \mu^2(\sigma_2-\mu\sigma_4) \\ \sigma_4 & 0 & 0 & 0 & 0 & \mu^2\sigma_6 & 0 \\ 0 & \mu^2\sigma_4 & a_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_2-\mu\sigma_4 & 0 & 0 & \mu^2\sigma_4 \end{bmatrix}.$$

Note that the other $O(\mu^2)$ terms in row 1 cannot be eliminated by a similar procedure — a simple sequence of row and column operations would only move the positions of these $O(\mu^2)$ terms, without replacing them by smaller quantities. For larger filter lengths, most $O(\mu^2)$ can be eliminated, and the above procedure proves to be useful.

We now apply the change of variables described in (4.14), obtaining the new matrix

$$\bar{\Phi}^{(3)} \triangleq \begin{bmatrix} \bar{a}_1 & 0 & 0 & 0 & 0 & \mu_f^2\bar{a}_2 & 0 \\ \bar{\sigma}_2 & 0 & 0 & 0 & 0 & \mu_f^2\bar{\sigma}_4 & 0 \\ 0 & \mu_f^2\bar{\sigma}_2 & \bar{a}_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{a}_1 & 0 & 0 & \mu_f^2\left(\bar{\sigma}_2 - \mu_f\bar{\sigma}_4\right) \\ \bar{\sigma}_4 & 0 & 0 & 0 & 0 & \mu_f^2\bar{\sigma}_6 & 0 \\ 0 & \mu_f^2\bar{\sigma}_4 & \bar{a}_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{\sigma}_2 - \mu_f\bar{\sigma}_4 & 0 & 0 & \mu_f^2\bar{\sigma}_4 \end{bmatrix},$$

where

$$\bar{a}_1 \triangleq 1 - 2\mu_f\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4, \qquad \bar{a}_2 \triangleq \bar{\sigma}_2 - 2\mu_f\bar{\sigma}_4 + \mu_f^2\bar{\sigma}_6.$$

We have thus found a sequence of similarity transformations such that the transformed matrix satisfies $\|\bar{\Phi}^{(3)}(\mu_f)\|_\infty < 1$ for small enough μ, and we can compute a bound for μ by solving

$$\bar{\mu} \triangleq \max_{\|\bar{\Phi}^{(3)}(\mu_f)\|_\infty < 1} \mu_f\mu_a,$$

where the condition $\left\|\bar{\Phi}^{(3)}\right\|_\infty < 1$ is equivalent to

$$\left|1 - 2\mu_f\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4\right| + \mu_f^2\left|\bar{\sigma}_2 - 2\mu_f\bar{\sigma}_4 + \mu_f^2\bar{\sigma}_6\right| < 1, \tag{4.15a}$$

$$\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4 < 1, \tag{4.15b}$$

$$\left|1 - 2\mu_f\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4\right| + \mu_f^2\bar{\sigma}_2 < 1, \tag{4.15c}$$

$$\left|1 - 2\mu_f\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4\right| + \mu_f^2\left|\bar{\sigma}_2 - \mu_f\bar{\sigma}_4\right| < 1, \tag{4.15d}$$

$$\bar{\sigma}_4 + \mu_f^2\bar{\sigma}_6 < 1, \tag{4.15e}$$

$$\left|\bar{\sigma}_2 - 2\mu_f\bar{\sigma}_4 + \mu_f^2\bar{\sigma}_6\right| + \mu_f^2\bar{\sigma}_4 < 1, \tag{4.15f}$$

$$\left|\bar{\sigma}_2 - \mu_f\bar{\sigma}_4\right| + \mu_f^2\bar{\sigma}_4 < 1. \tag{4.15g}$$

The key properties of Φ that allowed us to choose simple similarity transformations were:

1. The relative position of the entries 1, -2μ in the first row of Φ; and of the entries σ_2 and $-2\mu\sigma_2$ in the second row (and similarly with entries $1-\mu\sigma_2$ and $-\mu(1-2\mu\sigma_2)$ in row 4; and $\sigma_2 - \mu\sigma_4$ and $-\mu(\sigma_2 - 2\mu\sigma_4)$ in row 7).

2. The fact that only a few, very specific rows were modified by each row and column operation.

The relative position of the entries in rows 1 and 2 allowed us not only to replace the largest entry in row 1 with a smaller quantity, but also reduced all the other entries to be $O(\mu^2)$.

4.2 THE GENERAL STATE-SPACE MODEL

We now extend the earlier discussion to the case $M > 2$. As mentioned before, the size of Φ will grow exponentially fast with M. For example, for $M = 3$, Φ is already 37×37 (see Table 4.1). Also, some new properties of Φ arise that

were not present in the $M = 2$ case. Therefore, in the sequel we shall search for properties of Φ, and avoid actually forming it explicitly.

A state equation of the form (4.1) can be obtained for any filter of general length M, with

$$\boldsymbol{x}_k = \begin{bmatrix} a(k-M+1) & a(k-M+2) & \dots & a(k) \end{bmatrix}^T,$$

where (4.5) still holds. The starting point is again (4.6), from which we obtain expressions for each entry of $\mathrm{E}\,\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$ in terms of quantities available at time k. We saw in the $M = 2$ example that these expressions depend not only on the entries of $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$, but also on other quantities for which we also need to find recursions.

As we mentioned before, the total number of necessary variables grows exponentially fast with the filter length M. We therefore need to define a general notation to describe the linear system without actually forming the matrices.

We show in the appendix that all the entries in $\boldsymbol{\Gamma}_k$ are of the form

$$\mathrm{E}\left(x_{k,1}^{p_1} x_{k,2}^{p_2} \dots x_{k,M-1}^{p_{M-1}} \tilde{w}_{k,i} \tilde{w}_{k,j}\right), \tag{4.16}$$

where $1 \leq i, j \leq M$, and $0 \leq p_1, \dots, p_M \leq 4(M-1)$. In other words, each variable is the expected value of a product of two entries of $\tilde{\boldsymbol{w}}_k$ and powers of several entries of $\boldsymbol{x}_k$. Note also that powers of the last entry of $\boldsymbol{x}_k$, $x_{k,M}$, do not appear in the above definition. This is because $x_{k,M} = a(k)$ is independent of $\tilde{\boldsymbol{w}}_k$ and of all other $a(n)$, so we can compute expectations involving this term without having to define new variables, as the example below shows

$$\mathrm{E}\left(x_{k,1}^{p_1} \dots x_{k,M-1}^{p_{M-1}} x_{k,M}^{p_M} \tilde{w}_{k,i} \tilde{w}_{k,j}\right) = \mathrm{E}\left(a(k)^{p_M}\right) \mathrm{E}\left(x_{k,1}^{p_1} \dots x_{k,M-1}^{p_{M-1}} \tilde{w}_{k,i} \tilde{w}_{k,j}\right),$$

where $\mathrm{E}\big(a(k)^r\big) = \sigma_r$ (by definition).

To refer to (4.16), we shall use the notation

$$z^{i,j}_{\left(\begin{smallmatrix}1,\ p_1\\ \cdots\\ M-1,\ p_{M-1}\end{smallmatrix}\right)}(k) \triangleq \mathrm{E}\left(x_{k,1}^{p_1}\ldots x_{k,M-1}^{p_{M-1}}\tilde{w}_{k,i}\tilde{w}_{k,j}\right).$$

The superscript and subscript of the symbol on the left-hand side completely describe the variable. The superscript is a pair of integers listing the two entries of $\tilde{\boldsymbol{w}}_k$ that appear in the expectation. The subscript, which we refer to as the *power list* of the variable, is a list of pairs of integers. In a pair (i, p_i), the first integer (which we refer to as the *index* of the pair) represents an entry of $\boldsymbol{x}_k$ (i.e., $x_{k,i}$), and the second integer is the power to which $x_{k,i}$ is raised in the variable's definition. Accordingly, we refer to the second number of a pair as its *power*. Pairs that have zero power (*i.e.*, pairs $(i, 0)$) are usually omitted from the power list. For example, $z^{3,4}_{\left(\begin{smallmatrix}1,2\\ 5,3\end{smallmatrix}\right)}(k)$ represents the variable $\mathrm{E}\left(x_{k,1}^2 x_{k,5}^3 \tilde{w}_{k,3}\tilde{w}_{k,4}\right)$.

A variable with a subscript of 0 represents one of the entries of $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$, i.e.,

$$z_0^{i,j}(k) = \mathrm{E}\left(\tilde{w}_{k,i}\tilde{w}_{k,j}\right).$$

We shall refer to such variables as *seed variables*. All other variables (that is, variables with a nontrivial power list) will be referred to as *auxiliary variables*.

In Appendix 4.A we show that the recursions for each one of these seed variables can be obtained by inspection from only four basic recursions. We give a summary of these results here, and refer to the appendix for the details.

The four basic recursions are listed below. Note that $z_0^{i,j}(k) = z_0^{j,i}(k)$, so we assume without loss of generality that $i \leq j$.

1. $\underline{i = j < M}$

$$z_0^{i,i}(k+1) = z_0^{i,i}(k) - 2\mu\, z_{(i,\,2)}^{i,i}(k) + \mu^2\, z_{(i,\,4)}^{i,i}(k) - 2\mu \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,1}{m,\,1}}^{i,\,m}(k) +$$

$$+ 2\mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,3}{m,\,1}}^{i,\,m}(k) + \mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} \sum_{\substack{l=1 \\ l\neq i \\ l\neq m}}^{M-1} z_{\begin{pmatrix} i,\,2 \\ m,\,1 \\ l,\,1 \end{pmatrix}}^{m,\,l}(k) +$$

$$+ \mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,2}{m,\,2}}^{m,m}(k) + \mu^2\sigma_2\, z_{(i,\,2)}^{M,M}(k) - 2\mu\sigma_1\, z_{(i,\,1)}^{i,\,M}(k) +$$

$$+ 2\mu^2\sigma_1\, z_{(i,\,3)}^{i,\,M}(k) + 2\mu^2\sigma_1 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,2}{m,\,1}}^{m,\,M}(k). \tag{4.17}$$

2. $\underline{i = j = M}$

$$z_0^{M,M}(k+1) = \big(1 - 2\mu\sigma_2 + \mu^2\sigma_4\big)\, z_0^{M,M}(k) - 2\mu\big(\sigma_1 - \mu\sigma_3\big) \sum_{m=1}^{M-1} z_{(m,\,1)}^{m,\,M}(k) +$$

$$+ \mu^2\sigma_2 \sum_{m=1}^{M-1} \sum_{\substack{l=1 \\ l\neq m}}^{M-1} z_{\binom{m,\,1}{l,\,1}}^{m,\,l}(k) + \mu^2\sigma_2 \sum_{m=1}^{M-1} z_{(m,\,2)}^{m,m}(k).$$

$$\tag{4.18}$$

3. $\underline{i < j < M}$

$$z_0^{i,j}(k+1) = z_0^{i,j}(k) - \mu\ z_{(i,2)}^{i,j}(k) - \mu\ z_{(j,2)}^{i,j}(k) + \mu^2\ z_{\binom{i,2}{j,2}}^{i,j} - \mu \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\binom{j,1}{l,1}}^{l,i}(k) +$$

$$+ \mu^2 \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\begin{pmatrix} i,2 \\ j,1 \\ l,1 \end{pmatrix}}^{l,i}(k) - \mu\ z_{\binom{i,1}{j,1}}^{i,i}(k) + \mu^2\ z_{\binom{i,3}{j,1}}^{i,i}(k) - \mu \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\binom{i,1}{l,1}}^{l,j}(k) +$$

$$+ \mu^2 \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\begin{pmatrix} j,2 \\ i,1 \\ l,1 \end{pmatrix}}^{l,j}(k) - \mu\ z_{\binom{i,1}{j,1}}^{j,j}(k) + \mu^2\ z_{\binom{i,1}{j,3}}^{j,j}(k) + \mu^2 \sum_{\substack{l=1 \\ l\neq i \\ l\neq j}}^{M-1} z_{\begin{pmatrix} i,1 \\ j,1 \\ l,2 \end{pmatrix}}^{l,l}(k) + \tag{4.19}$$

$$+ \mu^2\sigma_2\ z_{\binom{i,1}{j,1}}^{M,M}(k) - \mu\sigma_1\ z_{(j,1)}^{i,M}(k) + 2\mu^2\sigma_1\ z_{\binom{i,2}{j,1}}^{i,M}(k) - \mu\sigma_1\ z_{(i,1)}^{j,M}(k) +$$

$$+ 2\mu^2\sigma_1\ z_{\binom{i,1}{j,2}}^{j,M}(k) + \mu^2 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} \sum_{\substack{m=1 \\ m\neq j \\ m\neq l}}^{M-1} z_{\begin{pmatrix} i,1 \\ j,1 \\ l,1 \\ m,1 \end{pmatrix}}^{l,m}(k) + 2\mu^2\sigma_1 \sum_{\substack{l=1 \\ l\neq i \\ l\neq j}}^{M-1} z_{\binom{j,1}{l,1}}^{l,M}(k).$$

4. $\underline{i < M}$

$$
\begin{aligned}
z_0^{i,M}(k+1) &= \big(1-\mu\sigma_2\big)\; z_0^{i,M}(k) - \mu\big(1-2\mu\sigma_2\big)\; z_{(i,2)}^{i,M}(k) - \\
&- \mu\big(1-2\mu\sigma_2\big) \sum_{\substack{l=1 \\ l\neq i}}^{M-1} z_{\binom{i,1}{l,1}}^{l,M}(k) - \mu\sigma_1 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} z_{(l,1)}^{i,l}(k) + \mu^2\sigma_1 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} z_{\binom{i,2}{l,1}}^{i,l}(k) - \\
&- \mu\sigma_1\; z_{(i,1)}^{i} + \mu^2\sigma_1\; z_{(i,3)}^{i,i}(k) - \mu\sigma_1\; z_{(i,1)}^{M,M}(k) + \\
&+ \mu^2\sigma_3\; z_{(i,1)}^{M,M}(k) + \mu^2\sigma_1 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} \sum_{\substack{m=1 \\ m\neq l}}^{M-1} z_{\binom{i,1}{m,1}}^{l,m}(k) + \mu^2\sigma_1 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} z_{\binom{i,1}{l,2}}^{l,l}(k).
\end{aligned}
\tag{4.20}
$$

For our $M = 2$ example, the above recursions reduce to:

1. $\underline{i = j = 1}$

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z_{(1,2)}^{1,1}(k) + \mu^2 z_{(1,4)}^{1,1} + \mu^2\sigma_2 z_{(1,2)}^{2,2}(k). \tag{4.21}$$

2. $\underline{i = j = 2}$

$$z_0^{2,2}(k+1) = \big(1 - 2\mu\sigma_2 + \mu^2\sigma_4\big) z_0^{M,M}(k) + \mu^2\sigma_2 z_{(1,2)}^{1,1}(k). \tag{4.22}$$

3. $\underline{i < j < 2}$: there is no such term.

4. $\underline{i < M}$

$$z_0^{1,2}(k+1) = \big(1 - 2\mu\sigma_2\big) z_0^{1,2}(k) - \mu(1 - 2\mu\sigma_2) z_{(1,2)}^{1,2}(k). \tag{4.23}$$

Note that we assumed $\sigma_3 = 0$, in order to be able to compare these recursions with those derived in Sec. 4.1.

The operation defined below translates the shift structure of the input sequence $\{\boldsymbol{x}_k\}$ to our z-notation, and will be useful to describe the recursions for auxiliary variables. Consider the generic power list

$$\mathcal{I} = \begin{pmatrix} 1, & p_1 \\ 2, & p_2 \\ & \cdots \\ M-1, & p_{M-1} \end{pmatrix}.$$

We define the new power list $\mathcal{I}+1$ as

$$\mathcal{I}+1 = \begin{pmatrix} 1, & 0 \\ 2, & p_1 \\ & \cdots \\ M-1, & p_{M-2} \end{pmatrix}. \tag{4.24}$$

The net effect is that the powers are shifted down in the power list. Note that the power of the index 1 in $\mathcal{I}+1$ is always zero, and that the power of $M-1$ in $\mathcal{I}$ (i.e., p_{M-1}) does not appear in $\mathcal{I}+1$.

With this operation, it is easy to write the recursion for a generic auxiliary variable $z_{\mathcal{I}}^{i,j}(k+1)$ using the four basic recursions. The algorithm below describes the procedure (using the case $i=j<M$ as an example).

Algorithm 4.1. *The recursion for an auxiliary variable $z_{\mathcal{I}}^{i,j}(k+1)$ can be obtained by the following modifications of the appropriate basic recursions* (4.17)–(4.20). *As we show in Appendix 4.A, two cases must be considered separately.*

(A) *The power of the index $M-1$ in $\mathcal{I}$ is zero.*

In this case, the expression for $z_{\mathcal{I}}^{i,j}(k+1)$ can be obtained from $z_0^{i,j}(k+1)$ by augmenting the power lists of all variables on the right-hand side of

(according to the values of i and j) (4.17)–(4.20) *with* $\mathcal{I}+1$. *For example, if* $i=j<M$, *we have*

$$z^{i,i}_{\mathcal{I}}(k+1) = z^{i,i}_{\mathcal{I}+1}(k) - 2\mu\, z^{i,i}_{\binom{i,2}{\mathcal{I}+1}}(k) + \mu^2\, z^{i,i}_{\binom{i,4}{\mathcal{I}+1}}(k) + \mu^2\sigma_2\, z^{M,M}_{\binom{i,2}{\mathcal{I}+1}}(k) + \ldots \tag{4.25}$$

Note that the power list $\binom{i,2}{\mathcal{I}+1}$ *may have repeated indices,* e.g., $\binom{i,2}{i,1}$. *This can be simplified by adding the powers of every instance of the repeated index. For example, the power list* $\binom{i,2}{i,1}$ *can be rewritten as* $(i,3)$.

(B) *The power of the index* $M-1$ *in* $\mathcal{I}$ *is* $p_{M-1}>0$.

In this case, in addition to augmenting the power lists of (4.17)–(4.20) *as before, we also perform the following substitutions:*

1. *Replace all* σ_r *appearing in the appropriate basic recursion* (4.17)–(4.20) *with* $\sigma_{r+p_{M-1}}$.
2. *Multiply by* $\sigma_{p_{M-1}}$ *all terms in the right-hand side of the resulting expression that are* not *multiples of any* σ_s *for* $s\geq 2$.

In the case $i=j<M$, *the result is*

$$z^{i,i}_{\mathcal{I}}(k+1) = \boxed{\sigma_{p_{M-1}}}\, z^{i,i}_{\mathcal{I}+1}(k) - 2\mu\boxed{\sigma_{p_{M-1}}}\, z^{i,i}_{\binom{i,2}{\mathcal{I}+1}}(k) + \\ + \mu^2\boxed{\sigma_{2+p_{M-1}}}\, z^{M,M}_{\binom{i,2}{\mathcal{I}+1}}(k) + \ldots$$

$\diamondsuit$

As an example, we apply this algorithm to obtain the recursions for the auxiliary variables in the $M=2$ example. We need the recursions for $z^{1,1}_{(1,2)}(k+1)$, $z^{1,1}_{(1,4)}(k+1)$, $z^{2,2}_{(1,2)}(k+1)$, and $z^{1,2}_{(1,2)}(k+1)$. The power of the index $M-1=1$ in

the power list of all these variables is nonzero, so we apply rule (B) for all these variables.

To obtain the recursion for $z_{(1,2)}^{1,1}(k+1)$, we apply (B) to (4.21), obtaining

$$z_{(1,2)}^{1,1}(k+1) = \sigma_2 z_0^{1,1}(k) - 2\mu\sigma_2 z_{(1,2)}^{1,1}(k) + \mu^2\sigma_2 z_{(1,4)}^{1,1}(k) + \mu^2\sigma_4 z_{(1,2)}^{2,2}(k).$$

The recursion for $z_{(1,4)}^{1,1}(k+1)$ is obtained in the same way:

$$z_{(1,4)}^{1,1}(k+1) = \sigma_4 z_0^{1,1}(k) - 2\mu\sigma_4 z_{(1,2)}^{1,1}(k) + \mu^2\sigma_4 z_{(1,4)}^{1,1}(k) + \mu^2\sigma_6 z_{(1,2)}^{2,2}(k).$$

Applying rule (B) to (4.22), we obtain

$$z_{(1,2)}^{2,2}(k+1) = \big(\sigma_2 - 2\mu\sigma_4 + \mu^2\sigma_6\big) z_0^{2,2}(k) + \mu^2\sigma_4 z_{(1,2)}^{1,1}(k).$$

The last recursion is obtained by the application of rule (B) to (4.23), resulting

$$z_{(1,2)}^{1,2}(k+1) = \big(\sigma_2 - \mu\sigma_4\big) z_0^{1,2}(k) - \mu\big(\sigma_2 - 2\mu\sigma_4\big) z_{(1,2)}^{1,2}(k).$$

Using our notation and the above method to obtain recursions for the variables in $\mathbf{\Gamma}_k$, we can deduce several properties of the state-space matrix Φ, as we show in the next section. However, before we proceed we must define some expressions that are used below.

1. The phrase "the recursion for the variable $z_{\mathcal{I}}^{i,j}(k+1)$" refers to the formula expressing $z_{\mathcal{I}}^{i,j}(k+1)$ in terms of variables at time k, *e.g.*, (4.25).

2. The N-th row (column) of Φ will be denoted by $\mathbf{r}\mathrm{ow}_N$ ($\mathbf{c}\mathrm{ol}_N$).

3. We define $\pi_{\mathcal{I}}^{i,j}$ as the position of variable $z_{\mathcal{I}}^{i,j}(k)$ in $\mathbf{\Gamma}_k$ (for example, $\pi_0^{1,1} = 1$ if $z_0^{1,1}(k)$ is the first entry of $\mathbf{\Gamma}_k$).

   The $\pi_{\mathcal{I}}^{i,j}$-th row of Φ describes recursion $z_{\mathcal{I}}^{i,j}(k+1)$, whereas the $\pi_{\mathcal{I}}^{i,j}$-th column of Φ contains the coefficients of $z_{\mathcal{I}}^{i,j}(k)$ in each recursion.

4.3 STRUCTURE OF Φ

Despite its large size, Φ is highly sparse and has considerable structure. Both features can be used to obtain approximations for some properties of LMS filters.

In [DP95], the sparseness of Φ was used to obtain approximations for its largest eigenvalue; nevertheless, the exponential growth of the length L of $\mathbf{\Gamma}_k$ limits the use of such sparse methods to filters of orders up to 6 or 7. To work with larger filter lengths, it is necessary to study the structure of Φ with more detail. In the remaining of this section, we derive several new properties of Φ that can be used to determine properties of the LMS filter without actually forming the matrix itself.

4.3.1 Sparsity of Φ

Although we shall not give an expression for L as a function of the filter length M, we can see that it grows exponentially with M from Table 4.1 and Fig. 4.1.

M	1	2	3	4	5	6
Length of $\mathbf{\Gamma}_k$ (L)	1	5	37	330	3,046	28,181
Number of non-zero elements per row of Φ	1	4	9	16	25	36

Table 4.1: *Size of* Φ *as a function of* M.

On the other hand, Algorithm 4.1 and recursions (4.A.6)–(4.A.9) imply that the number of nonzero elements in each row of Φ does not exceed $(M+1)^2$. Comparing this last number with the dimension of Φ in Table 4.1, we see that Φ

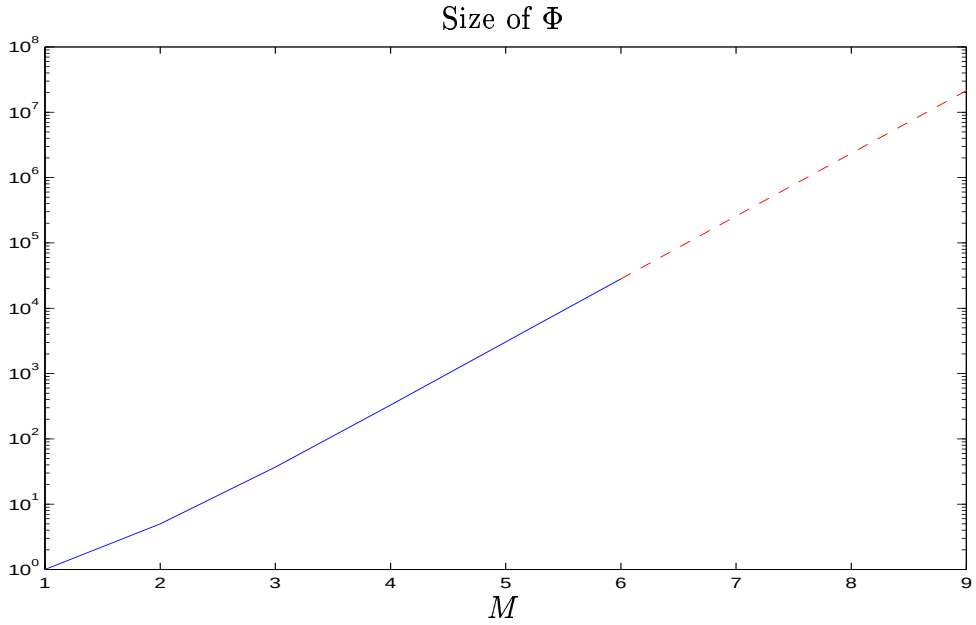

Figure 4.1: *Dimension of* Φ *for several values of* M*. Note that the vertical scale is logarithmic.*

is very sparse even for reasonably small M.

4.3.2 Permutations and Coefficient Sets

It is also possible to choose a small number of rows of Φ (no larger than $16M - 12$) such that *all other rows* of Φ are *permutations* of these rows. Consider the contrived example below (which does not correspond to any actual matrix Φ):

$$A = \begin{bmatrix} 1 & 0 & -2\mu & \mu^2\sigma_4 & 0 & 0 & 0 & -2\mu & 0 & 0 \\ 0 & -2\mu & 1 & \mu^2\sigma_4 & 0 & 0 & 0 & 0 & -2\mu & 0 \\ \sigma_2 & 0 & -2\mu\sigma_2 & \mu^2\sigma_6 & 0 & 0 & 0 & 0 & -2\mu\sigma_2 & 0 \\ 0 & -2\mu\sigma_2 & 0 & 0 & 0 & \mu^2\sigma_6 & \sigma_2 & 0 & 0 & -2\mu\sigma_2 \\ 0 & 1 & 0 & 0 & \mu^2\sigma_4 & 0 & 0 & -2\mu & 0 & -2\mu \end{bmatrix}.$$

The matrix A has two different classes of rows. Rows numbers 1, 2, and 5 form one class, and rows 3 and 4 form another class. In each class, the entries appearing in all rows are the same, but the order is different.

To describe a class of rows, we define its *coefficient set*, which is a list of the nonzero entries appearing in each row in the class. For example, for the class composed by rows 1, 2 and 5 above, the coefficient set is

$$\mathcal{A} = \Big\{(1,\, 1);\quad (-2\mu,\, 2);\quad (\mu^2\sigma_4,\, 1)\Big\}.$$

The pair $(-2\mu,\, 2)$ represents the fact that -2μ appears twice in each of the rows 1, 2 and 5.

Similarly, rows 3 and 4 are described by the set

$$\mathcal{B} = \Big\{(\sigma_2,\, 1);\quad (-2\mu\sigma_2,\, 2);\quad (\mu^2\sigma_6,\, 1)\Big\}.$$

We show in Appendix 4.B that a general Φ has a similar structure. In particular, all rows of the matrix take their nonzero entries from one of (at most) $16M - 12$ different coefficient sets. The coefficient sets of Φ are listed below (the sets are valid for $M > 3$), where p is an integer in the interval $0 \le p \le 4(M-1)$, and $\sigma_0 \triangleq 1$.

$$\begin{aligned}\Omega_p^1 = \Big\{&\big(\sigma_p,\, 1\big);\quad \big(-2\mu\sigma_p,\, M-1\big);\quad \big(\mu^2\sigma_{p+2},\, 1\big);\quad \big(2\mu^2\sigma_p,\, M-2\big);\\ &\big(\mu^2\sigma_p,\, (M-2)^2+1\big);\ \big(2\mu^2\sigma_{p+1},\, M-1\big)\Big\},\end{aligned} \tag{4.26a}$$

$$\begin{aligned}\Omega_p^M = \Big\{&\big(\sigma_p - 2\mu\sigma_{p+2} + \mu^2\sigma_{p+4},\, 1\big);\quad \big(-2\mu(\sigma_{p+1} - \mu\sigma_{p+3}),\, M-1\big);\\ &\big(\mu^2\sigma_{p+2},\, (M-1)^2\big)\Big\},\end{aligned} \tag{4.26b}$$

$$\begin{aligned}\Omega_p^{1,2} = \Big\{&\big(\sigma_p,\, 1\big);\quad \big(-\mu\sigma_p,\, 2(M-1)\big);\quad \big(\mu^2\sigma_p,\, (M-1)^2\big);\\ &\big(-\mu\sigma_{p+1},\, 2\big);\quad \big(2\mu^2\sigma_{p+1},\, M-1\big);\quad \big(\mu^2\sigma_{p+2},\, 1\big)\Big\},\end{aligned} \tag{4.26c}$$

$$\Omega_p^{1,\,M} = \Big\{ \big(\sigma_p - \mu\sigma_{p+2},\, 1\big);\quad \big(-\mu(\sigma_p - 2\mu\sigma_{p+2}),\, M-1\big) \\ \big(-\mu\sigma_{p+1},\, M\big);\quad \big(\mu^2\sigma_{p+1},\, (M-1)^2\big);\quad \big(\mu^2\sigma_{p+3},\, 1\big)\Big\}. \tag{4.26d}$$

4.3.3 Entries Identically Equal to 1

By an entry identically equal to one we mean an entry of Φ that is equal to one independently of the values of the step-size μ and of the input moments σ_p. These entries, which will play an important role in the stability analysis in Sec. 4.4, have the following property (the proof can be found in Appendix 4.C).

- There is at most a single entry equal to 1 in each row and column of Φ.

4.3.4 Properties of Seed Variables

Let $i \leq j < M$, and recall that the recursion for $z_0^{i,j}(k+1)$ is (we reproduce below the case $i = j$)

$$\begin{aligned} z_0^{i,i}(k+1) = z_0^{i,i}(k) - 2\mu\, z_{(i,\,2)}^{i,i}(k) + \mu^2\, z_{(i,\,4)}^{i,i}(k) - 2\mu \sum_{\substack{j=1\\ j\neq i}}^{M-1} z_{\binom{i,\,1}{j,\,1}}^{i,j}(k) + \\ + 2\mu^2 \sum_{\substack{j=1\\ j\neq i}}^{M-1} z_{\binom{i,\,3}{j,\,1}}^{i,j}(k) + \mu^2 \sum_{\substack{j=1\\ j\neq i}}^{M-1} \sum_{\substack{l=1\\ l\neq i\\ l\neq j}}^{M-1} z_{\binom{i,\,2}{j,\,1}}^{j,l}(k) + \mu^2 \sum_{\substack{j=1\\ j\neq i}}^{M-1} z_{\binom{i,\,2}{j,\,2}}^{j,j}(k) + \\ + \mu^2\sigma_2\, z_{(i,\,2)}^{M,M}(k) - 2\mu\sigma_1\, z_{(i,\,1)}^{i,\,M}(k) + 2\mu^2\sigma_1\, z_{(i,\,3)}^{i,\,M}(k) + \\ + 2\mu^2\sigma_1 \sum_{\substack{j=1\\ j\neq i}}^{M-1} z_{\binom{i,\,2}{j,\,1}}^{j,\,M}(k). \end{aligned}$$

This implies that the $\pi_0^{i,j}$-th row of Φ has an entry identically equal to 1 on the main diagonal. We show in Appendix 4.D that

- Only rows corresponding to seed variables can have an entry identically equal to 1 on the main diagonal,

- A seed variable $z_0^{i,j}(k)$ appears with a non-zero coefficent only in the recursions

$$z^{i,j}_{(M-1,\ p_{M-1})}(k+1)$$

with $p_{M-1} = 0$ or $2 \leq p_{M-1} \leq 4(M-1)$ (note that the superscripts are all the same). This implies that column $\mathbf{col}_{\pi_0^{i,j}}$ has nonzero entries only on positions $\pi^{i,j}_{(M-1,\ p_{M-1})}$, for p in the above range.

4.3.5 Block Structure of Φ

To each seed variable $z_0^{i,i}$ $(1 \leq i \leq M-1)$ there corresponds a dimension-$(M-i+1)$ block with the following pattern (we show an example of a dimension-4 block. Note that the dimension of a block is given by the number of rows that it contains):

$$B_4 \triangleq \begin{bmatrix} \boxed{1} & \boxed{-2\mu} & 0 & 0 & \mu^2\sigma_4 & -2\mu & 0 & 0 & \dots \\ 0 & 0 & \boxed{1} & 0 & 0 & 0 & -2\mu & 0 & \dots \\ 0 & 0 & 0 & \boxed{1} & 0 & 0 & 0 & -2\mu & \dots \\ \boxed{\sigma_2} & \boxed{-2\mu\sigma_2} & 0 & 0 & \mu^2\sigma_4 & -2\mu\sigma_2 & 0 & 0 & \dots \end{bmatrix}. \tag{4.27}$$

In general, a block will not consist of consecutive rows of Φ, but will be spread over the matrix. The important characteristic of each block is the position of the 1's on each row, as we now explain.

It is always possible to organize the rows and columns of Φ such that any given block of dimension K is placed in the first K rows of Φ, and such that element $(1,1)$ is 1, with element $(K,1)$ equal to σ_2, and such that the entries in

the upper superdiagonal in rows $2, 3, \ldots, K-1$ are equal to 1. However, since we are not forming the matrix, we describe a block in terms of the relative positions of the entries that are equal to 1 in Φ.

Let the rows of a size-K block correspond to rows r_1 to r_K in Φ. Then the first row of the block, r_1 (corresponding to recursion $z_0^{i,i}(k+1)$ with $i = M-K+1$), has a 1 on the main diagonal. Row r_K, which corresponds to $z_{(M-1,\,2)}^{i,i}(k+1)$, has all of its nonzero elements on the same columns as row r_1, but a σ_n in row r_1 is replaced by a σ_{n+2} in row r_K (i.e., the elements of row r_K belong to Ω_2^1, while the elements of row r_1 belong to Ω_0^1). The other rows are found as follows. Row r_{K-1} has a one in column r_K, row r_{K-2} has a 1 in column r_{K-2}, and so on.

This structure, and the relative position of the boxed entries in a block which are a consequence of the shift structure of the input sequence $\{\boldsymbol{x}_k\}$, allow us to analyze the stability of the LMS algorithm without actually forming Φ.

In addition to the $M-1$ blocks described above, Φ has $M(M-1)/2+1$ blocks corresponding to recursions $z_0^{i,j}(k+1)$ for $1 \le i < j < M$, with dimension $(M-i+1)$ each. We show below an example of a dimension-4 block corresponding to $M = 5$, $i = 2$, $j = 4$.

$$C_{2,4} = \begin{bmatrix} \boxed{1} & \boxed{-\mu} & 0 & 0 & -\mu & 0 & 0 & \ldots \\ 0 & 0 & \boxed{1} & 0 & 0 & -\mu & 0 & \ldots \\ 0 & 0 & 0 & \boxed{1} & 0 & 0 & -\mu & \ldots \\ \boxed{\sigma_2} & \boxed{-\mu\sigma_2} & 0 & 0 & -\mu\sigma_2 & 0 & 0 & \ldots \end{bmatrix}. \tag{4.28}$$

Again, the important feature is the relative position of the boxed entries, and the chain of 1's in the upper superdiagonal.

A more complete description of the block structure of Φ can be found in Appendix 4.E.

4.4 MEAN-SQUARE STABILITY OF LMS

The goal of this section is to derive a bound for the largest step-size that guarantees a uniformly bounded covariance matrix $\mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T$, i.e., to find an approximation for $\mu_{\max}$ defined by (4.2). In view of the results in the previous section, the LMS algorithm will be stable if and only if $|\lambda\big(\Phi(\mu)\big)| < 1$.

The eigenvalues of $\Phi(\mu)$ depend nonlinearly on μ and, hence, it is mathematically intractable to derive an expression for the spectral radius of Φ in terms of μ. For this reason, we shall proceed via an alternative route.

For any $M \times M$ square matrix A, it holds that its spectral radius is upper bounded by its ∞-norm [HJ87], i.e.,

$$\rho(A) \leq \|A\|_\infty,$$

where $\|A\|_\infty$ is the maximum absolute row sum of A,

$$\|A\|_\infty = \max_{1\leq i\leq n}\left(\sum_{j=1}^{n}|A_{i,\,j}|\right) = \text{ maximum row sum.}$$

Therefore, if we can find a μ that guarantees $\|\Phi(\mu)\|_\infty < 1$, then it also guarantees $\rho\big(\Phi(\mu)\big) < 1$.

We mentioned in the previous section that the rows of Φ belong to $16M - 12$ classes of rows, and that each row has only $O(M^2)$ nonzero entries. Using these properties, $\|\Phi\|_\infty$ can be evaluated with little effort, since

$$\|\Phi\|_\infty = \max_{1\leq i\leq L}\left(\sum_{j=1}^{L}|\phi_{i,\,j}|\right),$$

where $\phi_{i,\,j}$ is the $(i,\,j)$-th entry of Φ.

Unfortunately, however, several of the $\phi_{i,\,j}$ are equal to 1, which makes $\|\Phi\|_\infty$ always larger than 1 regardless of μ. In addition, Φ contains several entries equal

to σ_p for $2 \leq p \leq 4(M-1)$, and it would be restrictive to require that $\sigma_p < 1$ for all p in the above range.

This last difficulty can be overcome with a simple change of variables that we describe in Appendix 4.F. Nevertheless, this change of variables does not modify the entries that are equal to 1. To deal with those entries, we propose to construct a similarity transformation T such that there exists a $\bar{\mu} > 0$ satisfying

$$\|T^{-1}\Phi(\bar{\mu})T\|_\infty < 1. \tag{4.29}$$

Now since similarity transformations preserve eigenvalues, we obtain

$$\rho\big(\Phi(\bar{\mu})\big) \leq \|T^{-1}\Phi(\bar{\mu})T\|_\infty < 1. \tag{4.30}$$

Still, the similarity transformation T must be chosen such that the infinity norm of $T^{-1}\Phi(\mu)T$ remains easily computable.

We describe the construction of the similarity transformation in Appendix 4.G. Here we will show what the final transformed matrix $T^{-1}\Phi T$ is, in terms of a block B_k, as in (4.27).

Recall that each block B_k has a row with a 1 on the main diagonal and $k-2$ rows with a 1 outside of the main diagonal. Our task is to replace these 1's with smaller terms. The final matrix will have the following characteristics:

1) Rows that have a 1 on the main diagonal are replaced by rows that have $1 - 2\mu\sigma_2 + \mu^2\sigma_4$ on the diagonal. All other elements are either $O(\mu^2\sigma_2)$ or $O(\mu^3)$.

2) Rows that have a 1 in off-diagonal positions are replaced by rows that have $O(\mu^{1/\tau})$ on these positions, where $\tau > 0$ is a parameter that will be optimized in (4.32). All other elements will be $O(\mu^{(\tau-1)/\tau})$, e.g.,

$$\begin{bmatrix} \boxed{1} & -2\mu & 0 & 0 & -2\mu & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \boxed{1} & 0 & 0 & -2\mu & -2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \boxed{1} & 0 & 0 & 0 & 0 & -2\mu & -2\mu \\ \sigma_2 & -2\mu\sigma_2 & 0 & 0 & -2\mu\sigma_2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

is replaced by

$$\begin{bmatrix} \boxed{1-2\mu\sigma_2+\mu^2\sigma_4} & 0 & 0 & 0 & -\mu^2\sigma_2 & \mu^3 & \mu^3 & 0 & \mu^3 & \mu^3 \\ 0 & 0 & \boxed{\mu^{1/\tau}} & 0 & 0 & -2\mu^{(\tau-1)/\tau} & -2\mu^{(\tau-1)/\tau} & 0 & 0 & 0 \\ 0 & 0 & 0 & \boxed{\mu^{1/\tau}} & 0 & 0 & 0 & 0 & -2\mu & -2\mu \\ \mu^{-1/\tau}\sigma_2 & 0 & 2\mu^{(\tau-1)/\tau}\sigma_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.31}$$

Once the matrix is put into this form, it is possible to compute an estimate $\bar{\mu} \leq \mu_{\max}$. The bound $\bar{\mu}$ is the largest value of μ such that $\|T^{-1}\Phi(\mu)T\|_\infty < 1$, and is obtained from (see also Appendix 4.F)

$$\boxed{\bar{\mu} \stackrel{\Delta}{=} \max_{\substack{\text{s.t. inequalities(4.33a) to} \\ \text{(4.33d) are satisfied}}} \mu_a\mu_f.} \tag{4.32}$$

The maximization is performed over $\mu_a > 0$, $\mu_f > 0$, and $\tau > 1$. The constraints are given below, where $0 \leq p \leq 4(M-1)$ (they are written in terms of $\bar{\sigma}_p \stackrel{\Delta}{=} \mu_a\sigma_p$)

$$\begin{aligned} &|1-2\mu_f\bar{\sigma}_2+\mu_f^2\bar{\sigma}_4| + \mu_f^2|\bar{\sigma}_2-2\mu_f\bar{\sigma}_4+\mu_f^2\bar{\sigma}_6| + 2(M-2)(M+3)\mu_f^2\bar{\sigma}_2 + \\ &+\frac{1}{2}(M-2)(M-1)(M^3-M^2-M-5)\mu_f^3 + 13(M-2)(M-1)\mu_f^3\bar{\sigma}_2 + \\ &+(M^2-5M+9)\mu_f^3\bar{\sigma}_4 + 2\mu_f^3\bar{\sigma}_5 + (M-2)(M-1)^2(M^3-3M^2+3M-7)\mu_f^4 + \\ &+\frac{1}{2}(M-2)(M^3+9M^2-11M-21)\mu_f^4\bar{\sigma}_2 + 8(M-1)(M-2)\mu_f^4\bar{\sigma}_3 + \\ &+2(M-2)\mu_f^3\bar{\sigma}_3 + (2M^3-2M^2-15M+23)\mu_f^4\bar{\sigma}_4 + 2(M-1)\mu_f^4\bar{\sigma}_5 < 1 \end{aligned} \tag{4.33a}$$

$$\mu_f^{\frac{1}{(M-2)\tau}} + 2(M-1)\mu_f^{\frac{\tau-1}{\tau}} + (M-1)^2\mu_f^{\frac{2\tau-1}{\tau}} + \mu_f^{\frac{2\tau-1}{\tau}}\bar{\sigma}_2 < 1, \tag{4.33b}$$

$$|1 - 2\mu_f\bar{\sigma}_2 + \mu_f^2\bar{\sigma}_4| + (M-1)^2\mu_f^2\bar{\sigma}_2 + 2(M-1)\mu_f^2\bar{\sigma}_3 < 1, \tag{4.33c}$$

$$\begin{aligned}\bar{\sigma}_p + (M-1)(M-2)\mu_f\bar{\sigma}_p + (2M^3 - 2M^2 - 11M + 16)\mu_f^2\bar{\sigma}_p + \\ +2\mu_f\bar{\sigma}_{p+1} + 2(M-1)\mu_f^2\bar{\sigma}_{p+1} + \mu_f^2\bar{\sigma}_{p+2} < \mu_f^{\frac{1}{\tau}}.\end{aligned} \tag{4.33d}$$

Each constraint corresponds to the absolute sum of one row of the transformed matrix.

Note that there is always a choice of (μ_f, μ_a) that simultaneously satisfies all the above inequalities. In fact, for any given μ_a, inequalities (4.33a)–(4.33c) can be satisfied by choosing a sufficiently small μ_f (for which only the terms in the dominant — i.e., smaller — power of μ_f must be considered). For example (4.33a) and (4.33c) can be replaced by

$$1 - 2\mu_f\bar{\sigma}_2 < 1$$

for $\mu_f \approx 0$. The left-hand side of inequality (4.33b) becomes close to zero for $\mu_f \approx 0$, so this inequality can also be satisfied by choosing μ_f sufficiently small.

On the other hand, for small μ_f inequality (4.33d) becomes simply (since $\tau > 1$, $\mu_f^{1/\tau} \gg \mu_f$ for $\mu_f \approx 0$)

$$\bar{\sigma}_p < \mu_f^{\frac{1}{\tau}},$$

which can always be satisfied by choosing μ_a small enough.

The solid curve in Figure 4.2 shows a plot of the solution $\bar{\mu}$ of (4.32) for a sequence $\{a(k)\}$ that is normally distributed with $\sigma_2 = 0.01$. The rate of decay of $\bar{\mu}$ is dependent on the signal distribution and is proportional to $1/M^4$ in this example, as indicated by the broken line. Although $\bar{\mu}$ is not a tight bound for $\mu_{\max}$ (the discussion in the end of Sec. 6.3 has some more comments on this), it is the *first* computable bound (that we are aware of) to be derived.

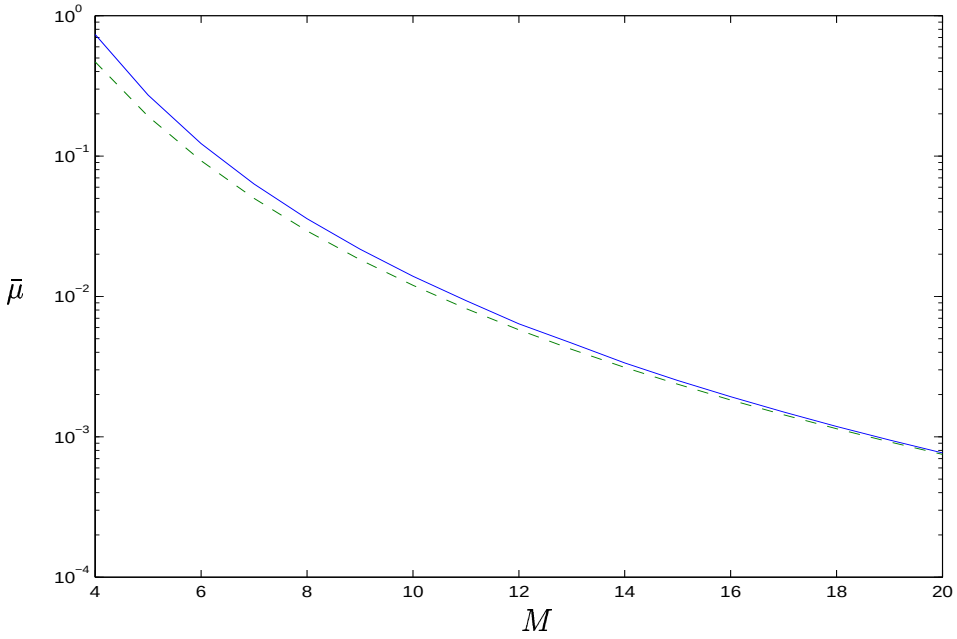

Figure 4.2: *A plot of $\bar{\mu}$ versus filter size M for a Gaussian distribution (continuous line), compared with the curve $1/(\sigma_2 M^4)$ (broken line).*

4.5 CONTRIBUTIONS OF THIS CHAPTER

In this chapter we described a new method to study the mean-square stability of LMS by means of a state-space model. The originality of our analysis is that it provides the first *computable* lower bound for the maximum step-size that guarantees mean-square stability of LMS, without assuming that the input sequence $\{\boldsymbol{x}_k\}$ is iid, and without assuming a-priori that the adaptation is "slow" (or that the step-size is infinitesimily small). It is therefore not a mere existence result (such as those presented in Sec. 3.2 and 3.3). Some of the results in this Chapter appeared in [NS98a].

As noted in [Slo93], the analysis of adaptive filters for larger step-sizes without independence assumptions is an important void in the literature. The only performance results published so far have been [FF86] and [DP95], but they suf-

fer from the large dimensions of the state-space matrix involved (since the full structure of these matrices is not exploited in these works as we do here).

There are several results stating that there is a $\mu_{\max}$ such that LMS and other adaptive algorithms will be stable for all step-sizes satisfying $0 < \mu < \mu_{\max}$ (for example, the results in Secs. 3.1 and 3.2 are of this kind, and prove the existance of a $\mu_{\max}$ as described above under certain assumptions on the correlation of the input sequence $\{\boldsymbol{x}_k\}$) [Maz79, JCR82, ME83]). Other existence results include [Bit83] and [BAN86, Sol97]. This last work provides a bound for the step-size that guarantees *almost-sure* stability of LMS. Nevertheless, not only that bound is not computable but, as we shall explain in Chapter 5, almost-sure stability does not imply mean-square stability or reasonable performance.

The analysis in this chapter also points to an interesting connection with the control literature, in special the theory of singular perturbations [KBB86]. In fact, it can be shown that a singularly-perturbed model for the Φ matrix for $M = 2$ in (4.10) reduces to the model used in the ODE analysis (see Appendix 4.H). It was already noted in [KY97] that ODE methods correspond to singular perturbation approximations, but until now there is no theory relating the full model (as in (4.1)) and the singularly perturbed models of ODE analysis.

APPENDICES FOR CHAPTER 4

4.A RECURSIONS FOR GENERIC VARIABLES

In this appendix, we describe in more detail the results of Sec. 4.2. In particular, we give the full expressions for the four basic recursions.

Consider the expression for $\tilde{w}_{k+1,i}^2$, obtained by expanding the (i,i) entry of $\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$ in (4.6):

$$\begin{aligned}\tilde{w}_{k+1,i}^2 = \big(1-2\mu x_{k,i}^2+\mu^2 x_{k,i}^4\big)\tilde{w}_{k,i}^2 - 2\mu\big(1-\mu x_{k,i}^2\big)x_{k,i}\sum_{j=2}^{M} x_{k,j}\tilde{w}_{k,j}\tilde{w}_{k,i} + \\ + \mu^2 x_{k,i}^2\left(\sum_{j=2}^{M}\sum_{\substack{l=2\\ l\neq j}}^{M} x_{k,j}x_{k,l}\tilde{w}_{k,j}\tilde{w}_{k,l} + \sum_{j=2}^{M} x_{k,j}^2\tilde{w}_{k,j}^2\right).\end{aligned} \tag{4.A.1}$$

Let $i=1$ and take expected values on both sides (recalling that $x_{k,M}=a(k)$ is independent of $\tilde{\boldsymbol{w}}_k$ and of $a(k-n)$, for all $n>0$), to obtain the recursion

$$\mathrm{E}\left(\tilde{w}_{k+1,1}^2\right) = \mathrm{E}\left(\tilde{w}_{k,1}^2\right) - 2\mu\,\mathrm{E}\left(x_{k,1}^2\tilde{w}_{k,1}^2\right) + \mu^2\,\mathrm{E}\left(x_{k,1}^4\tilde{w}_{k,1}^2\right) + \ldots \tag{4.A.2}$$

$\mathrm{E}\,\tilde{w}_{k+1,1}^2$ depends on expected values of products of entries of $\boldsymbol{x}_k$ and $\tilde{\boldsymbol{w}}_k$, with the general form $\mathrm{E}\left(x_{k,1}^{p_1}x_{k,j}^{p_j}x_{k,l}^{p_l}\tilde{w}_{k,l}\tilde{w}_{k,j}\right)$, for $1\leq j,l\leq M$. Recursions for each of these quantities must also be found.

For example, to obtain an expression for $\mathrm{E}\left(x_{k+1,1}^2\tilde{w}_{k+1,1}^2\right) = z_{(1,2)}^{1,1}(k+1)$, we proceed as follows. Recalling that $x_{k+1,1}=x_{k,2}$, multiply both sides of (4.A.1)

by $x_{k,2}^2$ to obtain

$$\begin{aligned} x_{k+1,1}^2\tilde{w}_{k+1,1}^2 &= x_{k,2}^2\big(1-2\mu x_{k,1}^2+\mu^2 x_{k,1}^4\big)\tilde{w}_{k,1}^2 - \\ &\quad - 2\mu x_{k,2}^2\big(1-\mu x_{k,1}^2\big)x_{k,1}\sum_{j=2}^{M} x_{k,j}\tilde{w}_{k,j}\tilde{w}_{k,1} + \\ &\quad + \mu^2 x_{k,1}^2 x_{k,2}^2\left(\sum_{j=2}^{M}\sum_{\substack{l=2\\ l\neq j}}^{M} x_{k,j}x_{k,l}\tilde{w}_{k,j}\tilde{w}_{k,l} + x_{k,2}^2\sum_{j=2}^{M} x_{k,j}^2\tilde{w}_{k,j}^2\right). \end{aligned} \tag{4.A.3}$$

Taking expectations, we obtain the recursion

$$\mathrm{E}\left(x_{k+1,1}^2\tilde{w}_{k+1,1}^2\right) = \mathrm{E}\left(x_{k,2}^2\tilde{w}_{k,1}^2\right) - 2\mu\,\mathrm{E}\left(x_{k,1}^2x_{k,2}^2\tilde{w}_{k,1}^2\right) + \mu^2\,\mathrm{E}\left(x_{k,1}^4x_{k,2}^2\tilde{w}_{k,1}^2\right) + \ldots \tag{4.A.4}$$

Note that the above expression can be obtained from (4.A.2) by putting the term $x_{k,2}^2$ inside every expectation on the right-hand side of (4.A.2), i.e., $\mathrm{E}\left(x_{k,1}^2\tilde{w}_{k,1}^2\right)$ in (4.A.2) is replaced by $\mathrm{E}\left(x_{k,1}^2x_{k,2}^2\tilde{w}_{k,1}^2\right)$ in (4.A.4). A similar observation applies for any variable

$$\mathrm{E}\left(x_{k+1,1}^{p_1}\ldots x_{k+1,M-2}^{p_{M-2}}x_{k+1,M-1}^{p_{M-1}}\tilde{w}_{k+1,i}\tilde{w}_{k+1,j}\right)$$

for which $p_{M-1}=0$. This observation explains Step (A) of Algorithm 4.1 in Sec. 4.2.

On the other hand, a variable $\mathrm{E}\left(x_{k+1,1}^{p_1}\ldots x_{k+1},x_{k+1,M-1}^{p_{M-1}}\tilde{w}_{k+1,1}^2\right)$ with $p_{M-1}>0$ has to be treated differently, since $x_{k+1,M-1}=x_{k,M}=a(k)$ is independent of $\tilde{\boldsymbol{w}}_k$. Step (B) of Algorithm 4.1 can be understood by repeating the arguments employed to obtain (4.A.4) to $z_{(M-1,2)}^{i,i}(k+1)$.

Algorithm 4.1 depends on the knowledge of the four basic recursions. We give the full expressions now.

4.A.1 Recursions for Generic Seed Variables

We can obtain the expressions for all seed variables at time $k+1$ in the same way as we obtained (4.A.2), *i.e.*, expanding the (i,j)-th entry of $\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$ and computing the expectation. The diagonal entries of $\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$ were computed in (4.A.1). For the off-diagonal entries we have

$$\begin{aligned}
\tilde{w}_{k+1,i}\tilde{w}_{k+1,j} &= \Big(1-\mu\big(x_{k,i}^2+x_{k,j}^2\big)+\mu^2 x_{k,i}^2 x_{k,j}^2\Big)\tilde{w}_{k,i}\tilde{w}_{k,j} - \\
&- \mu x_{k,j}\big(1-\mu x_{k,i}^2\big)\sum_{\substack{l=1\\ l\neq j\\ l\neq i}}^{M} x_{k,l}\tilde{w}_{k,l}\tilde{w}_{k,i} - \mu\big(1-\mu x_{k,i}^2\big)x_{k,j}x_{k,i}\tilde{w}_{k,i}^2 - \\
&- \mu x_{k,i}\big(1-\mu x_{k,j}^2\big)\sum_{\substack{l=1\\ l\neq i\\ l\neq j}}^{M} x_{k,l}\tilde{w}_{k,l}\tilde{w}_{k,j} - \mu\big(1-\mu x_{k,j}^2\big)x_{k,i}x_{k,j}\tilde{w}_{k,j}^2 + \\
&+ \mu^2 x_{k,i}x_{k,j}\sum_{\substack{l=1\\ l\neq i}}^{M}\sum_{\substack{m=1\\ m\neq j\\ m\neq l}}^{M} x_{k,l}x_{k,m}\tilde{w}_{k,l}\tilde{w}_{k,m} + \mu^2 x_{k,i}x_{k,j}\sum_{\substack{l=1\\ l\neq i\\ l\neq j}}^{M} x_{k,l}^2\tilde{w}_{k,l}^2.
\end{aligned} \tag{4.A.5}$$

Taking expectations of (4.A.1) and (4.A.5), we obtain:

1. $\underline{i = j < M}$:

$$z_0^{i,i}(k+1) = z_0^{i,i}(k) - 2\mu\, z_{(i,\,2)}^{i,i}(k) + \mu^2\, z_{(i,\,4)}^{i,i}(k) - 2\mu \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,1}{m,\,1}}^{i,\,m}(k) +$$

$$+ 2\mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,3}{m,\,1}}^{i,\,m}(k) + \mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} \sum_{\substack{l=1 \\ l\neq i \\ l\neq m}}^{M-1} z_{\binom{i,\,2}{m,\,1}}^{m,\,l}(k) +$$

$$+ \mu^2 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,2}{m,\,2}}^{m,m}(k) + \mu^2\sigma_2\, z_{(i,\,2)}^{M,M}(k) - 2\mu\sigma_1\, z_{(i,\,1)}^{i,\,M}(k) +$$

$$+ 2\mu^2\sigma_1\, z_{(i,\,3)}^{i,\,M}(k) + 2\mu^2\sigma_1 \sum_{\substack{m=1 \\ m\neq i}}^{M-1} z_{\binom{i,\,2}{m,\,1}}^{m,\,M}(k). \tag{4.A.6}$$

2. $\underline{i = j = M}$ — because $x_{k,M} = a(k)$ is independent of $\tilde{\boldsymbol{w}}_k$ and $a(k-n)$, we have a different expression in this case:

$$z_0^{M,M}(k+1) = \big(1 - 2\mu\sigma_2 + \mu^2\sigma_4\big)\, z_0^{M,M}(k) - 2\mu\big(\sigma_1 - \mu\sigma_3\big) \sum_{m=1}^{M-1} z_{(m,\,1)}^{m,\,M}(k) +$$

$$+ \mu^2\sigma_2 \sum_{m=1}^{M-1} \sum_{\substack{l=1 \\ l\neq m}}^{M-1} z_{\binom{m,\,1}{l,\,1}}^{m,\,l}(k) + \mu^2\sigma_2 \sum_{m=1}^{M-1} z_{(m,\,2)}^{m,m}(k). \tag{4.A.7}$$

3. $\underline{i \neq j,\ i < j < M}$:

$$z_0^{i,j}(k+1) = z_0^{i,j}(k) - \mu\ z_{(i,2)}^{i,j}(k) - \mu\ z_{(j,2)}^{i,j}(k) + \mu^2\ z_{\binom{i,2}{j,2}}^{i,j} - \mu \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\binom{j,1}{l,1}}^{l,i}(k) +$$

$$+ \mu^2 \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\begin{pmatrix} i,2 \\ j,1 \\ l,1 \end{pmatrix}}^{l,i}(k) - \mu\ z_{\binom{i,1}{j,1}}^{i,i}(k) + \mu^2\ z_{\binom{i,3}{j,1}}^{i,i}(k) - \mu \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\binom{i,1}{l,1}}^{l,j}(k) +$$

$$+ \mu^2 \sum_{\substack{l=1 \\ l\neq j \\ l\neq i}}^{M-1} z_{\begin{pmatrix} j,2 \\ i,1 \\ l,1 \end{pmatrix}}^{l,j}(k) - \mu\ z_{\binom{i,1}{j,1}}^{j,j}(k) + \mu^2\ z_{\binom{i,1}{j,3}}^{j,j}(k) + \mu^2 \sum_{\substack{l=1 \\ l\neq i \\ l\neq j}}^{M-1} z_{\begin{pmatrix} i,1 \\ j,1 \\ l,2 \end{pmatrix}}^{l,l}(k) + \tag{4.A.8}$$

$$+ \mu^2\sigma_2\ z_{\binom{i,1}{j,1}}^{M,M}(k) - \mu\sigma_1\ z_{(j,1)}^{i,M}(k) + 2\mu^2\sigma_1\ z_{\binom{i,2}{j,1}}^{i,M}(k) - \mu\sigma_1\ z_{(i,1)}^{j,M}(k) +$$

$$+ 2\mu^2\sigma_1\ z_{\binom{i,1}{j,2}}^{j,M}(k) + \mu^2 \sum_{\substack{l=1 \\ l\neq i}}^{M-1} \sum_{\substack{m=1 \\ m\neq j \\ m\neq l}}^{M-1} z_{\begin{pmatrix} i,1 \\ j,1 \\ l,1 \\ m,1 \end{pmatrix}}^{l,m}(k) + 2\mu^2\sigma_1 \sum_{\substack{l=1 \\ l\neq i \\ l\neq j}}^{M-1} z_{\binom{j,1}{l,1}}^{l,M}(k).$$

4. $\underline{i < j = M}$ — in this case, $x_{k,j} = a(k)$ is independent of the other variables,

and the above expression simplifies to:

$$z_0^{i,M}(k+1) = \big(1-\mu\sigma_2\big)\, z_0^{i,M}(k) - \mu\big(1-2\mu\sigma_2\big)\, z_{(i,2)}^{i,M}(k) -$$

$$- \mu\big(1-2\mu\sigma_2\big) \sum_{\substack{l=1\\ l\neq i}}^{M-1} z_{\binom{i,1}{l,1}}^{l,M}(k) - \mu\sigma_1 \sum_{\substack{l=1\\ l\neq i}}^{M-1} z_{(l,1)}^{i,l}(k) + \mu^2\sigma_1 \sum_{\substack{l=1\\ l\neq i}}^{M-1} z_{\binom{i,2}{l,1}}^{i,l}(k) -$$

$$- \mu\sigma_1\, z_{(i,1)}^{i}(k) + \mu^2\sigma_1\, z_{(i,3)}^{i,i}(k) - \mu\sigma_1\, z_{(i,1)}^{M,M}(k) + \tag{4.A.9}$$

$$+ \mu^2\sigma_3\, z_{(i,1)}^{M,M}(k) + \mu^2\sigma_1 \sum_{\substack{l=1\\ l\neq i}}^{M-1} \sum_{\substack{m=1\\ m\neq l}}^{M-1} z_{\binom{i,1}{m,1}}^{l,m}(k) + \mu^2\sigma_1 \sum_{\substack{l=1\\ l\neq i}}^{M-1} z_{\binom{i,1}{l,2}}^{l,l}(k).$$

4.B DETERMINING THE COEFFICIENT SETS

Each of the recursions (4.A.6)–(4.A.9) can be applied to several different seed variables, as we vary i and j (except for (4.A.7), which applies only to variable $z_0^{M,M}(k+1)$). Nevertheless, the coefficients appearing on the right-hand sides of (4.A.6)–(4.A.9) do not depend on the values of i and j. Therefore, even though there are $M(M+1)/2$ different seed variables (recall that $z_0^{i,j} = z_0^{j,i}$), there are only four different coefficient sets associated with seed variables (these coefficient sets correspond to (4.26) with $p = 0$).

The recursions for auxiliary variables are obtained from Algorithm 4.1. Note that Step (A) does not modify the coefficient sets, while Step (B) gives coefficient sets of form (4.26) for $p > 0$.

To conclude that the coefficient sets are as described in Sec. 4.3.2, we need to prove that p in (4.26) does not exceed $4(M-1)$. This follows from the lemma below.

Lemma 4.B.1. *There is no variable $z_{\mathcal{I}}^{i,j}(k)$ in $\mathbf{\Gamma}_k$ whose power list contains the pair $M-1, p_{M-1}$ with $p_{M-1} > 4(M-1)$.*

Proof: The proof is by induction. We first argue that there is no variable in $\mathbf{\Gamma}_k$ whose power list contains a pair $(1, p_1)$ with $p_1 > 4$.

Note that in all variables on the RHS of (4.A.6)–(4.A.9), the power lists are such that no index has power larger than 4. Since the power of the index 1 in $\mathcal{I}+1$ is *always* zero, steps (A) and (B) of Algorithm 4.1 imply that the generic recursion $z_{\mathcal{I}}^{i,j}(k+1)$ will also depend only on variables in whose power lists the index 1 has power no larger than 4. This means no variable in $\mathbf{\Gamma}_k$ can have a power list with a pair $(1, p_1)$ and $p_1 > 4$.

Consider again a generic recursion $z_{\mathcal{I}}^{i,j}(k+1)$. The above conclusion implies that the maximum power of the index 2 in $\mathcal{I}+1$ is 4, and repeating the above argument we conclude that there is no variable in $\mathbf{\Gamma}_k$ whose power list has a pair $(2, p_2)$ with $p_2 > 8$.

Proceeding with this argument, we conclude that the pair $(M-1, p_{M-1})$ in the power lists of all variables in $\mathbf{\Gamma}_k$ is such that $p_{M-1} \leq 4(M-1)$. This fact, and Step (B) in Algorithm 4.1, imply that the range of p in the coefficient sets (4.26) is $0 \leq p \leq 4(M-1)$.

$\diamondsuit$

This result also proves that the length of $\mathbf{\Gamma}_k$ is finite, and bounded by

$$\big(4\big)\big(2\times 4\big)\big(3\times 4\big)\big(\ldots\big)\big((M-1)\times 4\big) = 4^{M-1}(M-1)!$$

Although this is a loose upper bound for the length of $\mathbf{\Gamma}_k$, it shows that $\mathbf{\Gamma}_k$ and Φ are indeed finite.

4.C ENTRIES IDENTICALLY EQUAL TO 1

The row property of Sec. 4.3.3 can be verified by simple inspection of recursions (4.A.6)–(4.A.9). We prove the column property below, i.e., we show that every column of Φ has at most one entry equal to 1.

Assume that the same variable $z_{\mathcal{I}}^{i,j}(k)$ appears with a coefficient equal to 1 in the recursions for both $z_{\mathcal{J}}^{r,s}(k+1)$ and $z_{\mathcal{K}}^{m,n}(k+1)$,

$$z_{\mathcal{J}}^{r,s}(k+1) = z_{\mathcal{I}}^{i,j}(k) + \ldots, \qquad z_{\mathcal{K}}^{m,n}(k+1) = z_{\mathcal{I}}^{i,j}(k) + \ldots, \tag{4.C.1}$$

for some r, s, m, n, and where the power lists $\mathcal{J}$ and $\mathcal{K}$ are

$$\mathcal{J} = \begin{pmatrix} 1, & p_1, \\ \ldots & \\ M-1, & p_{M-1} \end{pmatrix}, \qquad \mathcal{K} = \begin{pmatrix} 1, & q_1, \\ \ldots & \\ M-1, & q_{M-1} \end{pmatrix},$$

for some integers $p_1, p_2, \ldots, p_{M-1}$ and $q_1, q_2, \ldots, q_{M-1}$.

We shall prove that the fact that $z_{\mathcal{I}}^{i,j}(k)$ appears with a coefficient of one in both recursions in (4.C.1) implies that

$$z_{\mathcal{J}}^{r,s}(k+1) \equiv z_{\mathcal{K}}^{m,n}(k+1), \tag{4.C.2}$$

i.e., that $r = m$, $s = n$, and $\mathcal{J} = \mathcal{K}$.

In fact, from recursions (4.A.6)–(4.A.9) and Algorithm 4.1, we must have

$$p_{M-1} = q_{M-1} = 0, \quad r = m = i, \quad s = n = j, \quad \text{and} \quad \mathcal{J} + 1 = \mathcal{K} + 1 = \mathcal{I}. \tag{4.C.3}$$

On the other hand, from (4.24), we need

$$p_1 = q_1, \qquad p_2 = q_2, \qquad \ldots \qquad p_{M-2} = q_{M-2} \tag{4.C.4}$$

in order to have $\mathcal{J} + 1 = \mathcal{K} + 1$.

Conditions (4.C.3) and (4.C.4) imply (4.C.2).

4.D PROPERTIES OF SEED VARIABLES

The properties described in Sec. 4.3.4 can be proved as follows.

Our first goal is to prove that only rows of Φ corresponding to recursions for seed variables $z_0^{i,j}(k+1)$ have an entry equal to 1 on the main diagonal.

A row of Φ has a constant entry 1 on the main diagonal if and only if the corresponding recursion $z_{\mathcal{I}}^{i,j}(k+1)$ is such that

$$z_{\mathcal{I}}^{i,j}(k+1) = z_{\mathcal{I}}^{i,j}(k) + \dots$$

However, from Algorithm 4.1 in Sec. 4.2, we know that the first term on the RHS should be $z_{\mathcal{I}+1}^{i,j}(k)$. Since the only case where $\mathcal{I}+1 = \mathcal{I}$ is $\mathcal{I} = 0$, it follows that only rows of Φ corresponding to seed variables have 1s on the main diagonal.

The second goal is to find in which recursions $z_{\mathcal{I}}^{r,s}(k+1)$ a seed variable $z_0^{i,j}(k)$ can appear with a nonzero coefficient. Assume then that $z_{\mathcal{I}}^{r,s}(k+1)$ depends on the seed variable $z_0^{i,j}(k)$. From Algorithm 4.1, we know that $z_{\mathcal{I}}^{r,s}(k+1)$ depends only on variables whose power list contains $\mathcal{I}+1$. Looking at the definition of $\mathcal{I}+1$, we see that the only case in which $\mathcal{I}+1 = 0$ is when $\mathcal{I} = (M-1,\ p_{M-1})$, for any $p_{M-1} \geq 0$.

4.E BLOCK STRUCTURE OF Φ

We describe how the blocks described in Sec. 4.3.5 are obtained using an example with block dimension 3 and $M = 4$. Consider the recursion for $z_0^{2,2}(k+1)$ (4.A.6),

$$z_0^{2,2}(k+1) = z_0^{2,2}(k) - 2\mu\ z_{(2,2)}^{2,2}(k) - \dots$$

On the other hand, the recursion for $z^{2,2}_{(2,2)}(k+1)$ is (applying Algorithm 4.1 to the recursion for $z^{2,2}_0(k+1)$ above)

$$z^{2,2}_{(2,2)}(k+1) = z^{2,2}_{(3,2)}(k) - 2\mu\, z^{2,2}_{\binom{2,2}{3,2}}(k) + \ldots$$

Finally, the recursion for $z^{2,2}_{(3,2)}(k+1)$ is

$$z^{2,2}_{(3,2)}(k+1) = \sigma_2\, z^{2,2}_0(k) - 2\mu\sigma_2\, z^{2,2}_{(2,2)}(k) + \ldots$$

Assume that $\pi^{2,2}_0 = 1$, $\pi^{2,2}_{(2,2)} = 2$, and $\pi^{2,2}_{(3,2)} = 3$. Then the first three rows of Φ form a block B_3

$$B_3 = \begin{bmatrix} 1 & -2\mu & 0 & \ldots \\ 0 & 0 & 1 & \ldots \\ \sigma_2 & -2\mu\sigma_2 & 0 & \ldots \end{bmatrix}.$$

Note that the dimension of the block is given by the number of rows that it contains, not the number of columns.

Following a similar procedure starting from any recursion $z^{i,j}_0(k+1)$ with $i = j < M$ ($i < j < M$), we obtain a block of form (4.27) (or (4.28)). The important point is that all recursions in the block correspond to variables with the same superscript as the initial seed variable (this property also ensures that each row appears at most in one single block).

There are actually two other kinds of blocks. One of them follows the pattern (shown below for a block with dimension 3)

$$D_3 = \begin{bmatrix} \boxed{1-\mu\sigma_2} & \boxed{-\mu} & 0 & 0 & \ldots \\ 0 & 0 & \boxed{1} & -\mu\sigma_2 & \ldots \\ \boxed{\sigma_2 - \mu\sigma_4} & \boxed{-\mu\sigma_2} & 0 & 0 & \ldots \end{bmatrix}. \qquad (4.E.1)$$

There are $M-1$ such blocks, corresponding to recursions $z^{i,M}_0(k+1)$ (for $1 \leq i \leq M-1$), with dimension $M-i+1$ each.

The pattern is similar to that of a block B_K (see Sec. 4.3.5), but instead of an entry 1 on the main diagonal, a block D_K has an entry $1-\mu\sigma_2$. The ones in the upper super-diagonal in the middle rows follow the same pattern as for a block B_K.

The last kind of block corresponds to recursions $z_{\mathcal{I}}^{i,j}(k+1)$ for which the power list $\mathcal{I}$ contains a pair $(1,p_1)$ with $p_1>0$. There is a large number of such blocks, with dimensions ranging from 2 to $M-1$. We show below a dimension-3 block as an example:

$$E_3 = \begin{bmatrix} 0 & \boxed{1} & 0 & 0 & -2\mu & 0 & 0 & 0 & \dots \\ 0 & 0 & \boxed{1} & 0 & 0 & -2\mu & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{\sigma_p} & -2\mu\sigma_p & \dots \end{bmatrix}. \tag{4.E.2}$$

It follows from Lemma 4.E.1 below that the column of Φ corresponding to the first column of E_3 has no entry equal to 1. This property guarantees that we cannot augment E_3 by introducing another row with a 1 in the first column. Therefore, the main characteristic of a block E_K is the absence of an entry equal to 1 or $1-\mu\sigma_2$ on the main diagonal.

Lemma 4.E.1. *An auxiliary variable $z_{\mathcal{I}}^{i,j}(k)$ whose power list contains the pair $(1,p_1)$ with $p_1>0$ does not appear in any recursion $z_{\mathcal{J}}^{r,s}(k+1)$ with a coefficient equal to 1 (i.e., a coefficient that is independent of the values of μ and of σ_p for any $p>0$).*

Proof: Assume that $p_1>0$ and that the variable $z^{i,j}_{\binom{1,\ p_1}{M-1,\ p_{M-1}}}(k)$ appears with a coefficient equal to 1 in the recursion $z_{\mathcal{J}}^{r,s}(k+1)$. From Algorithm 4.1, this recursion is given by

$$z_{\mathcal{J}}^{r,s}(k+1) = z_{\mathcal{J}+1}^{r,s}(k) + O(\mu),$$

Therefore, $z^{i,j}\left(\begin{smallmatrix}1,&p_1\\ \vdots&\\ M-1,&p_{M-1}\end{smallmatrix}\right)(k)$ can appear with a coefficient equal to 1 in the recursion $z_{\mathcal{J}}^{r,s}(k+1)$ only if $r=i$, $s=j$, and if

$$\mathcal{J}+1=\left(\begin{smallmatrix}1,p_1\\ \vdots\\ M-1,p_{M-1}\end{smallmatrix}\right).$$

But, by definition, $\mathcal{J}+1$ contains the pair $(1,0)$, which implies $p_1=0$, a contradiction.

$\diamondsuit$

4.F CHANGE OF VARIABLES

The condition $\|\Phi\|_\infty<1$ requires that the absolute row sum of rows with coefficient sets (4.26) be strictly less than one. For instance, assume that the i-th row of Φ has coefficient set (4.26) for some p satisfying $2\leq p\leq 4(M-1)$. Then the condition $\|\Phi\|_\infty<1$ requires that

$$\sum_{l=1}^{L}\left|(\Phi)_{i,l}\right|=\sigma_p+2\mu\sigma_p+\mu^2\sigma_p+2\mu\sigma_{p+1}+\cdots<1.$$

The above conditions require that $\sigma_p<1$ for $2\leq p\leq 4(M-1)$. This severe constraint on the input sequence can be relaxed via a simple change of variables, which we describe now. The idea is to split the step-size μ into two parts,

$$\mu=\mu_a\mu_f, \tag{4.F.1}$$

and redefine the input and desired sequences as

$$\bar{y}(k)\stackrel{\Delta}{=}\sqrt{\mu_a}\,y(k),\qquad \bar{\boldsymbol{x}}_k\stackrel{\Delta}{=}\sqrt{\mu_a}\,\boldsymbol{x}_k,\qquad \bar{a}(k)\stackrel{\Delta}{=}\sqrt{\mu_a}\,a(k).$$

These new definitions do not modify the LMS recursion, which is now written as

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu_f \bar{\boldsymbol{x}}_k \left(\bar{y}(k) - \boldsymbol{w}_k^T \bar{\boldsymbol{x}}_k\right).$$

Although the input variables and the step-size are different, the weight estimates are *not* modified by the change of variables.

The input moments are now given by

$$\bar{\sigma}_p \stackrel{\Delta}{=} \mathrm{E}\big(\bar{a}(k)^p\big) = \mu_a^{p/2}\sigma_p, \tag{4.F.2}$$

and we can make $\bar{\sigma}_p$ as small as we wish by choosing μ_a small enough. With this change of variables, (4.30) is replaced by the problem. Find a similarity transformation T such that

$$\bar{\mu} = \max_{\text{s.t. } \|T^{-1}\Phi(\mu_f)T\|_\infty < 1} \mu_a \mu_f,$$

where the minimization is performed over μ_a and μ_f. This leads to (4.32).

4.G SIMILARITY TRANSFORMATIONS

Our goal is to make each absolute row sum of the transformed matrix less than 1 for sufficiently small μ. To achieve this, we must modify all the elements $\phi_{i,\,j}$ that are equal to 1 so that they become dependent on μ, which is done differently depending on whether the 1 entry appears on the main diagonal or not.

A unit entry on the main diagonal appears in the first rows of blocks, as in B_n in (4.27) or $C_{m,n}$ in (4.28), while off-diagonal 1s appear on the other rows of these blocks, as well as in blocks D_n and E_n (4.E.1)–(4.E.2).

We assume here that the change of variables of Sec. 4.F has been performed, but to avoid excessive burden in the notation we write σ_p and μ instead of $\bar{\sigma}_p$ and μ_f.

We explain first how to construct similarity transformations that replace the diagonal 1s with $1 - 2\mu\sigma_2$, using a dimension 3 block B_3 as an example.

4.G.1 Transforming Diagonal Entries

The basic idea is to apply a sequence of elementary row and column operations, to replace the undesirable 1 on the main diagonal with $1 - 2\mu\sigma_2$. We shall first explain the basic construction, restricting our attention to the first 3 columns of a block B_3. Latter we show the impact of the transformations on the full matrix.

4.G.1.1 Basic Transformations

Consider the first 3 columns of a dimension-3 block of the type (4.27),

$$\bar{B}_3 = \begin{bmatrix} 1 & -2\mu & 0 \\ 0 & 0 & 1 \\ \sigma_2 & -2\mu\sigma_2 & 0 \end{bmatrix}.$$

This kind of pattern appears in Φ for $M > 2$. The first row of B_3 corresponds to the first coefficients in the recursion for $z_0^{i,i}(k+1)$, with $i = M - 2$. We shall present a sequence of similarity transformations that replaces the first row of $\bar{B}_3$ with a row whose absolute sum is strictly less than 1.

The basic idea is to subtract 2μ times the third row from the first row, so that the $(1,1)$ entry of $\bar{B}_3$ is replaced by $1 - 2\mu\sigma_2$. This can be accomplished by left-multiplying the block by the row operator

$$[I]_{1,3}^{-2\mu} \triangleq \begin{bmatrix} 1 & 0 & -2\mu \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Note that $[I]_{i,j}^{\alpha}$ is a modification of the identity matrix, with the (i,j)-th entry

substituted by α, and that its inverse is

$$\left([I]_{i,j}^{\alpha}\right)^{-1} = [I]_{i,j}^{-\alpha}.$$

Left-multiplying $\bar{B}_3$ by $[I]_{1,3}^{-2\mu}$ we obtain

$$\bar{B}_3^{(1)} \triangleq [I]_{1,3}^{-2\mu}\bar{B}_3 = \begin{bmatrix} 1-2\mu\sigma_2 & -2\mu(1-2\mu\sigma_2) & 0 \\ 0 & 0 & 1 \\ \sigma_2 & -2\mu\sigma_2 & 0 \end{bmatrix}.$$

To make this a similarity transformation, we need to post-multiply $\bar{B}_3^{(1)}$ by $[I]_{1,3}^{2\mu}$. This is a column operation that adds 2μ times the first column of $\bar{B}_{1,3}^{(1)}$ to its third column, resulting in

$$\bar{B}_3^{(2)} \triangleq [I]_{1,3}^{-2\mu}\bar{B}_3[I]_{1,3}^{2\mu} = \begin{bmatrix} 1-2\mu\sigma_2 & -2\mu(1-2\mu\sigma_2) & 2\mu\big(1-2\mu\sigma_2\big) \\ 0 & 0 & 1 \\ \sigma_2 & -2\mu\sigma_2 & 2\mu\sigma_2 \end{bmatrix}.$$

Although these operations do replace the constant entry 1 on the diagonal of $\bar{B}_3$ by a quantity that decreases with μ, the above matrix is still not useful for stability analysis, since there is no region around $\mu = 0$ for which the absolute sum of the first row is less than 1. Indeed, for $\mu < 1/(2\sigma_2)$ we have

$$\big|1-2\mu\sigma_2\big| + 4\mu\big|1-2\mu\sigma_2\big| = 1 + 2\mu(2-\sigma_2) - 8\mu^2\sigma_2.$$

As we saw in Sec. 4.F the absolute sum of the third row requires $\sigma_2 < 1$, which implies that $(2-\sigma_2) > 0$. Therefore, the above row sum is larger than one for $\mu < (2-\sigma_2)/(4\mu\sigma_2)$.

We can obtain a row sum smaller than 1 by applying similarity transformations that eliminate the $O(\mu)$ terms in the second and third columns of $\bar{B}_3^{(2)}$. The $O(\mu)$ term in the $(1,2)$ entry of $\bar{B}_3^{(2)}$ can be eliminated if we add 2μ times

the first column of $\bar{B}_3^{(2)}$ to its second column, i.e., post-multiply $\bar{B}_3^{(2)}$ by $[I]_{1,2}^{-2\mu}$ (and also pre-multiply the resulting matrix by $[I]_{1,2}^{2\mu}$ to complete the similarity transformation):

$$\bar{B}_3^{(3)} = [I]_{1,2}^{2\mu} \bar{B}_3^{(2)} [I]_{1,2}^{-2\mu} = \begin{bmatrix} 1 - 2\mu\sigma_2 & 0 & -4\mu^2 \\ 0 & 0 & 1 \\ \sigma_2 & 0 & 2\mu\sigma_2 \end{bmatrix}.$$

With this transformation, we successfully eliminated the $O(\mu)$ entries in $(1,2)$ and $(1,3)$ positions of $\bar{B}_3^{(2)}$, and the absolute sum of the first row now becomes

$$1 - 2\mu\sigma_2 + 4\mu^2 < 1 \quad \text{for} \quad \mu < \frac{\sigma_2}{2}.$$

4.G.1.2 Impact on the Full Matrix

To apply the above similarity transformations to the full matrix Φ, augment the elementary row and column operators so that they have the same dimension as Φ, i.e.,

$$[I]_{1,3}^{-2\mu} = \left[\begin{array}{ccc|cc} 1 & 0 & -2\mu & 0 & \dots \\ 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & \dots \\ \hline 0 & 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{array}\right].$$

We shall discuss how other elements of Φ are modified by the row and column operations, assuming (without loss of generality) that B_3 is the leading block in Φ.

Let us first consider what happens to the other columns of B_3. In fact, we

have

$$B_3 = \begin{bmatrix} 1 & -2\mu & 0 & -2\mu & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & -2\mu & 0 & \dots \\ \sigma_2 & -2\mu\sigma_2 & 0 & -2\mu\sigma_2 & 0 & 0 & \dots \end{bmatrix},$$

and after all the transformations of Sec. 4.G.1.1 are applied, we obtain

$$B_3^{(3)} = \begin{bmatrix} 1-2\mu\sigma_2 & 0 & -4\mu^2 & -2\mu(1-2\mu\sigma_2) & 4\mu^2 & 0 & \dots \\ 0 & 0 & 1 & 0 & -2\mu & 0 & \dots \\ \sigma_2 & 0 & 2\mu\sigma_2 & -2\mu\sigma_2 & 0 & 0 & \dots \end{bmatrix}.$$

Therefore, even after the similarity transformations of Sec. 4.G.1.1, the absolute sum of the first row of Φ is still lower bounded by 1, because of entries $-2\mu(1-2\mu\sigma_2)$, as the one in the $(1,4)$ position (in fact, there are several such entries). They can also be eliminated by sequences of row and column operations. Instead of continuing with our B_3 example, we will describe these operations for a generic length-M filter as an example.

We need to find transformations for each recursion $z_0^{i,j}(k+1)$ with $i < j \leq M$ and $i = j < M$. We will show that, not only the transformations for each recursion do not interfere with one another in any way, but they are in fact all very similar. Therefore, we will only describe in detail the operations that are applied to $z_0^{1,1}(k+1)$.

We seek a similarity transformation $\bar{\Phi} = T_{1,1}^{-1}\Phi T_{1,1}$, or, equivalently, a change of variables $\bar{\mathbf{\Gamma}}_k = T_{1,1}^{-1}\mathbf{\Gamma}_k$ that transforms the recursion

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) + O(\mu)$$

to

$$\bar{z}_0^{1,1}(k+1) = (1-2\mu\sigma_2)\bar{z}_0^{1,1}(k) + O(\mu^2).$$

We will construct $T_{1,1}$ via a sequence of simple similarity transformations S_n. In the description below, after each transformation S_n is applied, we have a new set of variables

$$\mathbf{\Gamma}_k^{(n)} = S_n^{-1}\mathbf{\Gamma}_k^{(n-1)}, \qquad \text{with} \qquad \mathbf{\Gamma}_k^{(0)} = \mathbf{\Gamma}_k.$$

Nevertheless, to avoid adding yet another index to our notation, we shall adopt the following convention: if the variable $z_{\mathcal{I}}^{i,j}(k)$ is the l-th entry of $\mathbf{\Gamma}_k$, we shall refer to the l-th entry of all $\mathbf{\Gamma}_k^{(n)}$ *always* by $z_{\mathcal{I}}^{i,j}(k)$ (and similary, the recursion described by the l-th row of every intermediate matrix will be referred to as $z_{\mathcal{I}}^{i,j}(k+1)$). These conventions greatly simplify the description of the similarity transformations.

Assume, without loss of generality, that the first row of Φ corresponds to the recursion $z_0^{1,1}(k+1)$. We shall describe now the full sequence of operations necessary to transform this row, i.e., we shall present a sequence of transformations such that the $(1,1)$ entry of the transformed matrix is $1-2\mu\sigma_2$ and all other entries of the first row are $O(\mu^2)$. The following recursions will come into play:

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu_f z_{1,2}^{1,1}(k) - 2\mu_f \sum_{j=2}^{M-1} z^{1,j}_{\binom{1,1}{j,1}}(k) + O(\mu^2), \tag{4.G.1a}$$

$$z_{1,2}^{1,1}(k+1) = z_{2,2}^{1,1}(k) - 2\mu z_{2,2}^{1,1}(k) - 2\mu \sum_{j=2}^{M-1} z^{1,j}_{\binom{1,1}{2,2}_{j,1}}(k) + O(\mu^2),$$

$$\vdots$$

$$z_{M-1,2}^{1,1}(k+1) = \sigma_2 z_0^{1,1}(k) - 2\mu\sigma_2 z_{1,2}^{1,1}(k) - 2\mu\sigma_2 \sum_{j=2}^{M-1} z^{1,j}_{\binom{1,1}{j,1}}(k) + O(\mu^2), \tag{4.G.1b}$$

$$z^{1,j}_{\binom{1,1}{j,1}}(k+1) = z^{1,j}_{\binom{2,1}{j+1,1}}(k) + O(\mu),$$
$$\vdots \qquad (4.G.1c)$$
$$z^{1,j}_{\binom{M-j,1}{M-1,1}}(k+1) = \sigma_1 z^{1,j}_{M-j+1,1}(k) + O(\mu),$$

where $j = 2 \ldots M-1$.

Algorithm 4.G.1. *Assume that the variables are ordered such that the first row of* Φ *corresponds to* $z_0^{1,1}(k+1)$*, the second to* $z^{1,1}_{(1,2)}(k+1)$*, etc, until the* M*-th row, which corresponds to* $z^{1,1}_{(M-1,2)}(k+1)$*. Assume also that the* $\beta_1^{(j)}$*th row corresponds to* $z^{1,j}_{\binom{1,1}{j,1}}(k+1)$*, that the* $\beta^{(j)}_{M-j}$*th corresponds to* $z^{1,j}_{\binom{M-j,1}{M-1,1}}(k+1)$ *and so on. The objective of the transformations is to modify the first row of* Φ*.*

Begin the transformations by eliminating the term

$$-2\mu z^{1,M-1}_{\binom{1,1}{M-1,1}}(k)$$

from (4.G.1a)*, which is accomplished by the following steps:*

(1.1) *Column operation.* *Add* $2\mu\,\mathbf{col}_1$ *to* $\mathbf{col}_{\beta_1^{(M-1)}}$*. This step zeros the* -2μ *element in the* $(1,\, \beta_1^{(M-1)})$ *position of* Φ*. After this transformation, the recursion* (4.A.6) *for* $z_0^{1,1}(k+1)$ *becomes*

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z^{1,1}_{1,2}(k) - 2\mu \sum_{j=2}^{M-2} z^{1,j}_{\binom{1,1}{j,1}}(k) + O(\mu^2). \qquad (4.G.2)$$

The only difference from (4.G.1a) *is that the sum now goes only up to* $M-2$*, instead of up to* $M-1$*.*

(1.2) *Row operation.* *Subtract* $2\mu\,\mathbf{row}_{\beta_1^{(M-1)}}$ *from* $\mathbf{row}_1$*, to complete the similarity transformation. Since* $z^{1,M-1}_{\binom{1,1}{M-1,1}}(k+1)$ *contains only* $O(\mu)$ *terms, only the* $O(\mu^2)$ *terms in* (4.G.2) *are modified by this step (see last recursion* (4.G.1c)*).*

These steps are equivalent to applying the transformation $S_{M-1} = [I]^{-2\mu}_{1,\beta_1^{(M-1)}}$ *to* Φ*:*

$$\Phi \leftarrow \left(S_{M-1}^{(1)}\right)^{-1} \Phi S_{M-1}^{(1)},$$

where the arrow is used to indicate that the same symbol is being used for the original and for the transformed Φ*. The column operation (1.1) corresponds to the right multiplication* ΦS_{M-1}*, and the row operation (1.2) corresponds to the left multiplication* $\left(S_{M-1}^{(1)}\right)^{-1}\left(\Phi S_{M-1}^{(1)}\right)$*.*

The elimination of the next elements is more involved. The next step eliminates

$$-2\mu z^{1,\,M-2}_{\binom{1,\,1}{M-2,\,1}}(k)$$

.

(2.1) Column operation. *Add* $2\mu\,\mathbf{col}_1$ *to* $\mathbf{col}_{\beta_1^{(M-2)}}$*. This step zeros the* -2μ *element in the* $(1,\,\beta_1^{(M-2)})$ *position of* Φ*. Now* (4.G.2) *reads*

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z_{1,\,2}^{1,1}(k) - 2\mu \sum_{j=2}^{M-3} z^{1,\,j}_{\binom{1,\,1}{j,\,1}}(k) + O(\mu^2). \qquad (4.G.3)$$

(2.2) Row operation. *To complete the similarity transformation, subtract* 2μ $\mathbf{row}_{\beta_1^{(M-2)}}$ *from* $\mathbf{row}_1$*. Unlike what happened in the previous case,* $\mathbf{row}_{\beta_1^{(M-2)}}$ *has a 1 in the position* $(\beta_1^{(M-2)},\,\beta_2^{(M-2)})$*. The row operation therefore creates a new element* -2μ *in the* $(1,\,\beta_2^{(M-2)})$ *position, so*

$$\begin{aligned} z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z_{1,\,2}^{1,1}(k) - 2\mu \sum_{j=2}^{M-3} z^{1,\,j}_{\binom{1,\,1}{j,\,1}}(k) - \\ - 2\mu z^{1,\,M-2}_{\binom{2,\,1}{M-1,\,1}}(k) + O(\mu^2). \end{aligned} \qquad (4.G.4)$$

(2.3) Follow steps (1.1)–(1.2) to eliminate this new element. Note that the new row operation does not add another $O(\mu)$ element to $\mathbf{row}_1$*, and* (4.G.4) *becomes*

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z_{1,2}^{1,1}(k) - 2\mu \sum_{j=2}^{M-3} z_{\binom{1,1}{j,1}}^{1,j}(k) + O(\mu^2). \qquad (4.G.5)$$

The transformations in steps (2.1)–(2.2) correspond to the similarity transformation $S_{M-2}^{(1)} = [I]_{1,\beta_1^{(M-2)}}^{2\mu}$*, and step (2.3) corresponds to* $S_{M-2}^{(2)} = [I]_{1,\beta_2^{(M-2)}}^{2\mu}$.

The elimination of each one of the $O(\mu)$ *elements in the* $(1, \beta_1^{(j)})$ *position can therefore be accomplished with* $M-j$ *similarity transformations. The first* $M-j-1$ *transformations follow (2.1)–(2.2), and the last one follows (1.1)–(1.2). After these transformations, the recursion* (4.G.5) *becomes*

$$z_0^{1,1}(k+1) = z_0^{1,1}(k) - 2\mu z_{1,2}^{1,1}(k) + O(\mu^2). \qquad (4.G.6)$$

All similarity transformations are of the form $S_{M-j}^{(l)} = [I]_{1,\beta_l^{(M-j)}}^{2\mu}$.

The last element to be eliminated is the -2μ *in position* $(1,\ 2)$ *(i.e.,* $-2\mu z_{1,2}^{1,1}(k)$ *in* (4.G.6)*). To zero this term, it is necessary to apply* $M-2$ *transformations (2.1)–(2.2), and one transformation (1.1)–(1.2). The only difference now is that after* $2\mu\,\mathbf{row}_M$ *is subtracted from* $\mathbf{row}_1$*, the final recursion is*

$$z_0^{1,1}(k+1) = (1 - 2\mu\sigma_2) z_0^{1,1}(k) + O(\mu^2), \qquad (4.G.7)$$

which is in the desired form. The similarity transformations have all the form $W^{(l)} = [I]_{1,n}^{2\mu}$*, for* $2 \le n \le M$.

$\diamondsuit$

All these steps together correspond to the application of a similarity transformation to Φ. The matrix describing the combined effect of all transformations

for $z_0^{i,i}(k+1)$ is denoted by

$$T_{(1,1)} = W^{(M-1)} W^{(M-2)} \dots W^{(1)} S_2^{(M-2)} \dots S_{M-1}^{(1)}. \tag{4.G.8}$$

Therefore, after all these steps, Φ is replaced with

$$\Phi \leftarrow T_{1,1}^{-1} \Phi T_{1,1}.$$

Although these steps are meant to modify the row of Φ corresponding to $z_0^{1,1}(k+1)$, the same basic idea can be used (almost with no modifications) to modify the rows of Φ corresponding to all other $z_0^{i,j}(k+1)$. The column and row operations that we used to define our similarity transformations have the following important properties, that we state as a lemma.

Lemma 4.G.1. *The similarity transformations used in Algorithm 4.G.1 are such that*

1. *The row operations modify only the first row of* Φ.
2. *Each column operation replaces a target column (say,* $\mathbf{c}\mathrm{ol}_n$*) by* $\mathbf{c}\mathrm{ol}_n + \alpha\, \mathbf{c}\mathrm{ol}_1$. *Such operation is performed on* $\mathbf{c}\mathrm{ol}_n$ only once.
3. *The column operations modify only rows that describe recursions* $z_{(M-1,p)}^{1,1}(k+1)$.

Proof: The first two properties follow directly from the definitions of the row and column operations. The third property follows from the remarks in Sec. 4.3.4, and from property 2.

$\diamondsuit$

These results hold in general — the transformations for the block starting with $z_0^{i,j}(k+1)$ will modify only rows relative to recursions $z_{M-1,p}^{i,j}(k+1)$. This

observation implies that the transformations for a block will not interfere in any way with the transformations for any other block. We state this result as another lemma.

Lemma 4.G.2. *Define $\mathcal{R}_0^{i,j}$ as the set of rows modified or used to modify other rows in the transformations for a block starting with $z_0^{i,j}(k+1)$. Then the intersection of $\mathcal{R}_0^{i,j}$ and $\mathcal{R}_0^{r,s}$ is empty if $r \neq i$ or $s \neq j$.*

♢

This observation allows us to apply the transformations for all blocks B_n (4.27), $C_{m,n}$ (4.28) and D_n (4.E.1) independently — one set of transformations will not interfere with the other. In particular, the transformations may be applied in any desired order. However, the diagonal transformations used in Sec. 4.G.2 below must be applied only after the transformations described in this section.

4.G.1.3 Second Level of Transformations

After the above transformations are applied to Φ, we can use a procedure similar to Algorithm 4.G.1 to eliminate the $O(\mu^2)$ terms in the transformed recursion $z_0^{1,1}(k+1)$ ($O(\mu^2\sigma_2)$ terms are not eliminated). This second level of transformations will put the matrix in the form described in (4.31).

These transformations follow the same basic idea as Algorithm 4.G.1 — the elimination of an $O(\mu^2)$ term from row $\pi_0^{1,1}$ requires a series of up to M row and column operations, exactly as described in steps (1.1)–(2.2) of Algorithm 4.G.1. In the process, the $1-2\mu\sigma_2$ entry in the main diagonal in row $\pi_0^{1,1}$ will be replaced by $1 - 2\mu\sigma_2 + \mu^2\sigma_4$.

The only significant difference between the second level of transformations and the first one is that not all $O(\mu^2)$ terms can be eliminated. This is because the recursion $z_0^{1,1}(k+1)$ (after transformation) contains the terms $\mu^2(1-2\mu\sigma_2)z^{j,j}_{\binom{1,2}{j,2}}(k)$, for $2 \leq j \leq M-1$. Consider for example the term $z^{M-1,M-1}_{\binom{1,2}{M-1,2}}(k)$. The recursions of interest are:

$$z_0^{1,1}(k+1) = (1-2\mu\sigma_2)z_0^{1,1}(k) + \mu^2(1-2\mu\sigma_2)z^{M-1,M-1}_{\binom{1,2}{M-1,2}}(k) + \ldots$$

$$z^{M-1,M-1}_{\binom{1,2}{M-1,2}}(k+1) = \sigma_2 z_{2,2}^{M-1,M-1}(k) + O(\mu)$$

If we subtract μ^2 times column $\pi_0^{1,1}$ from column $\pi^{M-1,M-1}_{\binom{1,2}{M-1,2}}$ to eliminate the undesirable term in row $\pi_0^{1,1}$, the row operation that completes the similarity transformations adds a term $\mu^2\sigma_2 z_{2,2}^{M-1,M-1}(k)$ to recursion $z_0^{1,1}(k+1)$. Therefore, we only replaced an $O(\mu^2)$ term by an $O(\mu^2\sigma_2)$ term.

However, almost all $O(\mu^2)$ terms in row $\pi_0^{1,1}$ will be of the form $\mu^2(1-2\mu\sigma_2)z^{1,l}_{\binom{1,3}{l,1}}(k)$, i.e., the power list contains an index with power 1. This fact allows us to replace an $O(\mu^2)$ term with an $O(\mu^3)$ term. For example, consider the elimination of the following term from $z_0^{1,1}(k+1)$:

$$z_0^{1,1}(k+1) = (1-2\mu\sigma_2)z_0^{1,1}(k) + 2\mu^2(1-2\mu\sigma_2)z^{1,M-1}_{\binom{1,3}{M-1,1}}(k) + \ldots$$

$$z^{1,M-1}_{\binom{1,3}{M-1,1}}(k+1) = O(\mu).$$

Therefore, the elimination of the undesired term in $z_0^{1,1}(k+1)$ will only add $O(\mu^3)$ entries to $\pi_0^{1,1}$.

The application of a slight modification of Algorithm 4.G.1 to all $O(\mu^2)$ entries of $z_0^{1,1}(k+1)$ whose power lists contain a pair $(j,1)$ eliminates the great majority of $O(\mu^2)$ entries in $\pi_0^{1,1}$, which allows for a bound $\bar{\mu}$ much less conservative than what would be obtained by eliminating only $O(\mu)$ terms.

4.G.2 Transforming Off-Diagonal Entries

Even after applying all the above transformations to modify the recursion $z_0^{i,j}(k+1)$, there still are several entries of Φ equal to 1 in off-diagonal positions. These entries can be replaced with μ^r for $r \in (0,\, 1)$ using diagonal transformations, as we explain now. Again, we present the transformations using the $z_0^{1,1}(k+1)$ block as an example:

$$\begin{aligned}
z_0^{1,1}(k+1) &= (1-2\mu\sigma_2)z_0^{1,1}(k) + O(\mu^2) \\
z_{(1,\,2)}^{1,1}(k+1) &= z_{(2,\,2)}^{1,1}(k) - 2\mu z_{\binom{1,\,2}{2,\,2}}^{1,1}(k) - \dots \\
z_{(2,\,2)}^{1,1}(k+1) &= z_{(3,\,2)}^{1,1}(k) - 2\mu z_{\binom{1,\,2}{3,\,2}}^{1,1}(k) - \dots \\
&\vdots \\
z_{(M-1,\,2)}^{1,1}(k+1) &= \sigma_2 z_0^{1,1}(k) + 2\mu\sigma_2 z_{(2,\,2)}^{1,1}(k) + \dots + 2\mu\sigma_2 z_{(M-1,\,2)}^{1,1}(k) + \dots
\end{aligned} \tag{4.G.9}$$

Assume, without loss of generality, that the vector $\mathbf{\Gamma}_k$ is ordered such that

$$\mathbf{\Gamma}_k = \begin{bmatrix} z_0^{1,1}(k) & z_{(1,\,2)}^{1,1}(k) & z_{(2,\,2)}^{1,1}(k) & \dots & z_{(M-1,\,2)}^{1,1}(k) & \times & \dots & \times \end{bmatrix}^T,$$

where $\times$ is any variable. Then the first elements of Φ are

$$\Phi = \begin{bmatrix} 1-2\mu\sigma_2+\mu^2\sigma_4 & \times & \times & \times & \dots & \times & \times & \times & \dots \\ 0 & 0 & 1 & \times & \dots & \times & \times & \times & \dots \\ 0 & 0 & 0 & 1 & \dots & \times & \times & \times & \dots \\ \vdots & & & \vdots & & & & \vdots & \\ 0 & 0 & 0 & 0 & \dots & 1 & \times & \times & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & \times & \dots \\ \sigma_2 & \times & \times & \times & \dots & \times & \times & \times & \dots \\ \times & \times & \times & \times & \dots & \times & \times & \times & \dots \\ \vdots & & & & \vdots & & & \vdots & \end{bmatrix},$$

where $\times$ stands for a possibly nonzero entry.

The 1 in Φ's $(2,3)$ position can be replaced with $\mu^{\frac{1}{(M-2)\tau}}$, by using the diagonal matrix $D_1 = \mathrm{diag}(1,\ 1,\ \mu^{\frac{1}{(M-2)\tau}},\ 1,\ \dots 1)$. The parameter $\tau > 0$ will not be chosen in advance. Instead, it will be used as an extra degree of freedom in the maximization (4.32).

Applying the above diagonal transformation, we obtain the new matrix

$$D_1^{-1}\bar{\Phi}D_1 =$$

$$= \begin{bmatrix} 1-2\mu\sigma_2+\mu^2\sigma_4 & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \dots & \times & \times & \times & \dots \\ 0 & 0 & \mu^{\frac{1}{(M-2)\tau}} & \times & \dots & \times & \times & \dots \\ 0 & 0 & 0 & \mu^{-\frac{1}{(M-2)\tau}} & \dots & \mu^{-\frac{1}{(M-2)\tau}}\times & \mu^{-\frac{1}{(M-2)\tau}}\times & \dots \\ \vdots & & & \vdots & & & & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & \times & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & \dots \\ \sigma_2 & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \times & \dots & \times & \times & \dots \\ \times & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \times & \dots & \times & \times & \dots \\ \vdots & & & & \vdots & & & \vdots \end{bmatrix}.$$

Thus, although the $(2,3)$ element was successfully replaced by $\mu^{\frac{1}{(M-2)\tau}}$, the element $(3,4)$ is now $\mu^{-\frac{1}{(M-2)\tau}}$. Now using $D_2 = \mathrm{diag}(1,\ 1,\ 1,\ \mu^{\frac{2}{(M-2)\tau}},\ 1,\ \dots,\ 1)$,

we obtain

$$\begin{bmatrix}
1-2\mu\sigma_2+\mu^2\sigma_4 & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \times & \times & \dots & \times & \times & \dots \\
0 & 0 & \mu^{\frac{1}{(M-2)\tau}} & \times & \times & \dots & \times & \times & \dots \\
0 & 0 & 0 & \mu^{\frac{1}{(M-2)\tau}} & \mu^{-\frac{1}{(M-2)\tau}}\times & \dots & \mu^{-\frac{1}{(M-2)\tau}}\times & \mu^{-\frac{1}{(M-2)\tau}}\times & \dots \\
0 & 0 & 0 & 0 & \mu^{-\frac{2}{(M-2)\tau}} & \dots & \mu^{-\frac{2}{(M-2)\tau}}\times & \mu^{-\frac{2}{(M-2)\tau}}\times & \dots \\
\vdots & & & \vdots & & & & & \vdots \\
0 & 0 & 0 & 0 & 0 & \dots & \times & \times & \dots \\
0 & 0 & 0 & 0 & 0 & \dots & 1 & \times & \dots \\
\sigma_2 & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \mu^{\frac{2}{(M-2)\tau}}\times & \times & \dots & \times & \times & \dots \\
\times & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \mu^{\frac{2}{(M-2)\tau}}\times & \times & \dots & \times & \times & \dots \\
\vdots & & & & \vdots & & & & \vdots
\end{bmatrix}.$$

Continuing this procedure, the result will be

$$\begin{bmatrix}
1-2\mu\sigma_2+\mu^2\sigma_4 & \times & \mu^{\frac{1}{(M-2)\tau}}\times & \dots & \mu^{\frac{M-3}{(M-2)\tau}}\times & \mu^{\frac{1}{\tau}}\times & \times & \dots \\
0 & 0 & \mu^{\frac{1}{(M-2)\tau}} & \dots & \mu^{\frac{M-3}{(M-2)\tau}}\times & \mu^{\frac{1}{\tau}}\times & \times & \dots \\
0 & 0 & 0 & \dots & \mu^{\frac{M-4}{(M-2)\tau}}\times & \mu^{\frac{M-3}{(M-2)\tau}}\times & \mu^{-\frac{1}{(M-2)\tau}}\times & \dots \\
0 & 0 & 0 & \dots & \mu^{\frac{M-5}{(M-2)\tau}}\times & \mu^{\frac{M-4}{(M-2)\tau}}\times & \mu^{-\frac{2}{(M-2)\tau}}\times & \dots \\
\vdots & & & \vdots & & & & \vdots \\
0 & 0 & 0 & \dots & \mu^{\frac{1}{(M-2)\tau}} & \mu^{\frac{2}{(M-2)\tau}}\times & \mu^{-\frac{M-4}{(M-2)\tau}}\times & \dots \\
0 & 0 & 0 & \dots & 0 & \mu^{\frac{1}{(M-2)\tau}} & \mu^{-\frac{M-3}{(M-2)\tau}}\times & \dots \\
\mu^{-\frac{1}{\tau}}\sigma_2 & \mu^{-\frac{1}{\tau}}\times & \mu^{\frac{M-3}{(M-2)\tau}}\times & \dots & \mu^{\frac{1}{(M-2)\tau}}\times & \times & \mu^{-\frac{1}{\tau}}\times & \dots \\
\mu^{-\frac{1}{\tau}}\times & \mu^{-\frac{1}{\tau}}\times & \mu^{\frac{M-3}{(M-2)\tau}}\times & \dots & \mu^{\frac{1}{(M-2)\tau}}\times & \times & \mu^{-\frac{1}{\tau}}\times & \dots \\
\vdots & & \vdots & & & & \vdots &
\end{bmatrix}.$$

This procedure must be repeated for all blocks B_n, $C_{m,n}$, D_n and E_n. The column property in Sec. 4.3.3 (stating that there is at most one entry equal to 1 on each column) guarantees that the above method can be applied to substitute all off-diagonal 1's in Φ with powers of the step-size.

4.G.3 Final Coefficient Sets and Stability Bound

We can track the results of each transformation on the coefficient sets (4.26). Although there will be more different coefficient sets than the $16M-12$ that

we started with, an analysis of Algorithm 4.G.1 and Sec. 4.G.2 shows that the transformed recursions of the largest blocks have the largest absolute row-sums after all transformations are applied. Therefore, the coefficient sets that must be considered for stability analysis are those of transformed recursions

$$z_0^{1,1}(k+1), \qquad z_{(i,2)}^{1,1}(k+1), \quad \text{for } 1 \le i \le M-1,$$
$$z_{(M-1,p)}^{1,1}(k+1), \quad \text{for } 1 \le p \le 4(M-1),$$

as well as the recursions

$$z_{(M-1,r)}^{M,M}(k+1), \quad \text{for } 0 \le r \le 4(M-1).$$

The absolute row-sums relative to these rows are the ones listed in (4.33).

4.H SINGULAR PERTURBATIONS AND THE ODE MODEL

We have seen in Chapter 3, Secs 3.1–3.3, that when the step-size is small ($\mu \approx 0$), results from independence theory are good approximations for the mean-square behavior of adaptive filters.

Another approach for the analysis of adaptive filters with slow adaptation (small step-size) is the ODE (from *ordinary differential equation*) method, which studies convergence with probability one. This method approximates the convergence characteristics of an adaptive filter via a continuous-time model [BMP87, KY97]. For example, the ODE approximation for $\mathrm{E}\,\tilde{\boldsymbol{w}}_k$ of LMS is a differential equation of the form

$$\dot{\boldsymbol{a}} = R\boldsymbol{a}, \tag{4.H.1}$$

where $R = \mathrm{E}\, \boldsymbol{x}_k \boldsymbol{x}_k^T$. The ODE method states that, for sufficiently small step-size (and under a number of conditions), the trajectory of $\tilde{\boldsymbol{w}}_k$ will stay close to that of $\boldsymbol{a}$.

On the other hand, an approximation method that is commonly used in the control literature is the singular perturbations method [KBB86, KKJ86]. This method allows us to find approximations and conditions for stability of dynamical systems, when a parameter approaches zero. For example, if a discrete-time linear system can be put into the form

$$\boldsymbol{b}_{k+1} = \begin{bmatrix} I + \epsilon A_{1,1} & \epsilon A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \boldsymbol{b}_k, \tag{4.H.2}$$

then (if $\epsilon \approx 0$ and under certain stability conditions), the vector $\boldsymbol{b}_k$ can be split into *slow* and *fast* subsystems, and the trajectory of the slow subsystem can be approximated by the continuous-time dynamical system

$$\dot{\boldsymbol{s}} = \left[A_{1,1} + A_{1,2} \big(I - A_{2,2} \big)^{-1} A_{2,1} \right] \boldsymbol{s}. \tag{4.H.3}$$

Although the theory of singular perturbations for discrete-time systems is not as well developed as that for continuous-time systems, there are results in the literature analyzing conditions on ϵ and on the $A_{i,j}$ for which the original system (4.H.2) is stable [SOK84, LL92, KA96].

The ODE method can be related to the singular perturbations method as follows. Consider again the LMS algorithm with $M = 2$, with the input sequence $\{\boldsymbol{x}_k\}$ satisfying the same conditions as in Sec. 4.1.1. We can then derive a state-space description for the entries of $\mathrm{E}\, \tilde{\boldsymbol{w}}_k$, in the same way as in the derivation of

Φ (from Eq. (4.10)), obtaining

$$\begin{bmatrix} \mathrm{E}\big(\tilde{w}_{k+1,1}\big) \\ \mathrm{E}\big(\tilde{w}_{k+1,2}\big) \\ \mathrm{E}\big(x_{k+1,1}^2 \tilde{w}_{k+1,1}\big) \end{bmatrix} = \left[\begin{array}{cc|c} 1 & 0 & -\mu \\ 0 & 1-\mu\sigma_2 & 0 \\ \hline \sigma_2 & 0 & -\mu\sigma_2 \end{array}\right] \begin{bmatrix} \mathrm{E}\big(\tilde{w}_{k,1}\big) \\ \mathrm{E}\big(\tilde{w}_{k,2}\big) \\ \mathrm{E}\big(x_{k,1}^2 \tilde{w}_{k,1}\big) \end{bmatrix}. \qquad (4.\mathrm{H}.4)$$

Let $A_{2,2} = [\,-\mu\sigma_2\,]$, and

$$\boldsymbol{\Theta}_k \triangleq \begin{bmatrix} \mathrm{E}\big(\tilde{w}_{k,1}\big) \\ \mathrm{E}\big(\tilde{w}_{k,2}\big) \\ \mathrm{E}\big(x_{k,1}^2 \tilde{w}_{k+1,1}\big) \end{bmatrix}, \quad A_{1,1} = \begin{bmatrix} 0 & 0 \\ 0 & -\sigma_2 \end{bmatrix}, A_{1,2} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, A_{2,1} = \begin{bmatrix} \sigma_2 & 0 \end{bmatrix}.$$

Now, applying the singular perturbations method to the above system, we obtain an approximation for the evolution of the first two entries of $\boldsymbol{\Theta}_k$ from the continuous-time system below:

$$\dot{\boldsymbol{\Theta}}_s = \begin{bmatrix} -\frac{\sigma_2}{1+\mu\sigma_2} & 0 \\ 0 & -\sigma_2 \end{bmatrix} \boldsymbol{\Theta}_s,$$

which for sufficiently small μ can be further approximated by $\dot{\boldsymbol{\Theta}}_s = -\sigma_2 I \boldsymbol{\Theta}_s$, which is the same model obtained from the ODE method (4.H.1).

In addition, if we apply the singular perturbations formula (4.H.3) to Φ, we obtain the slow subsystem (again for $\mu \approx 0$),

$$\dot{\boldsymbol{\Gamma}}_s = -2\sigma_2 \boldsymbol{\Gamma}_s,$$

which would be an ODE model for the entries of $\mathrm{E}\,\tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T$ [for $M = 2$, $\boldsymbol{\Gamma}_s$ has size 4].

With this connection, we can employ results from the theory of singular perturbations to study how good the approximations provided by the ODE method are, and also to determine a range of step-sizes for which we can guarantee stability of an adaptive filter.

CHAPTER 5

ENSEMBLE-AVERAGE LEARNING CURVES

Ensemble-average learning curves are commonly used to analyze and demonstrate the performance of adaptive filters. They are obtained by averaging several error curves over repeated experiments and by plotting the resulting average curve. Such averaged curves have been used to extract, among other things, information about the rate of convergence of an adaptive filter, its steady-state error value, or choices of step-sizes for faster or slower convergence. For infinitesimal step-sizes, or under the independence conditions of Chapter 2, it is known that data extracted from such ensemble-average learning curves provide reasonably accurate information about the real performance of an adaptive filter (see the discussion in Chapters 2 and 3, and [WML76, ME83, WS85, Ber86, Mac95, SK95, Hay96, KV96]).

In Chapter 4, however, we were interested in studying the performance of adaptive schemes for larger step-sizes and without independence assumptions. By larger step-sizes we do not mean step-sizes that are necessarily large, but rather step-sizes that are not infinitesimally small. In the process of comparing results obtained from ensemble-average learning curves with results predicted by an exact theoretical analysis for such scenarios, we noticed a considerable difference between both cases, and the differences persisted no matter how many more experiments we averaged. A first explanation was to blame the simulation program and possible numerical errors. After careful study, however, we realized

that the differences have an analytical explanation and that they do occur for larger step-sizes. Even more importantly, this led us to observe some other interesting phenomena regarding the behavior of ensemble-average learning curves that may have gone unnoticed in the literature. More specifically, and among other original results, we shall establish the following facts *both* by theory and by simulation:

1. Ensemble-average learning curves actually exhibit two distinct rates of convergence; one for the initial time instants and another for later time instants.

2. Ensemble-average learning curves tend to converge faster than predicted by theory.

3. Ensemble-average learning curves can (and do) converge even when a mean-square stability analysis predicts divergence.

4. The more experiments we average to construct an ensemble-average learning curve, the more time it takes to observe the distinction between theory and simulation. Nevertheless, (in the noiseless case) the difference always exists and one should only simulate for a longer period of time to observe it.

5. Mean-square analysis may not be the most appropriate performance measure for larger step-sizes. A combination of mean-square and almost-sure stability results seem to be more appropriate.

6. For filters with multiple taps, the behavior of ensemble-average learning curves may be dependent on the initial condition. This dependency does not exist for single tap filters.

5.1 LEARNING CURVES

We first recall the definitions of learning curves and ensemble-average learning curves. Recall that the MSE is defined as

$$\mathrm{E}\, e(k)^2 = \mathrm{E}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k\big)^2,$$

and the mean-square deviation (MSD), by

$$\mathrm{E}\, \|\boldsymbol{w}_* - \boldsymbol{w}_k\|^2 \ .$$

We assume in this section that:

M-1. *The sequences* $\big\{y(k),\, \boldsymbol{x}_k\big\}$ *are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$*, and where* $v(k)$ *is zero-mean with variance* σ_v^2 *and uncorrelated with* $\boldsymbol{x}_k$*.*

As we saw in Sec. 1.2, the LMS algorithm computes approximations to $\boldsymbol{w}_*$ via the recursion

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \boldsymbol{x}_k\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k\big).$$

The plot of the MSE as a function of the time instant k is known as the *learning curve* of the (LMS) algorithm, and it is dependent on the step-size μ. In general, it is not a simple task to find analytical expressions for the learning curve or for the steady-state MSE, except when the assumptions of independence theory are used. Despite the fact that these assumptions are seldom satisfied in practice, it is known that the learning curves obtained using independence theory are good approximations for the true learning curves when the step-size μ is vanishingly small (see Chapter 3).

5.1.1 Ensemble-Average Learning Curves

Unfortunately, however, there is no general result predicting how large will the difference be between the actual learning curve for a particular application, and the approximation obtained from independence theory, for a given value of the step-size. For this reason, it is common practice to estimate the learning curve by experimentation or by repeated simulations. More specifically, several independent experiments (or simulations) are performed, say L of them. In each of the experiments, the LMS algorithm is applied for N iterations, always starting from the same initial condition and under the same statistical conditions for the sequences $\{y(k)\}$ and $\{\boldsymbol{x}_k\}$. From each experiment i, a sample curve $e^{(i)}(k)$, $1 \leq k \leq N$, is obtained. After all L experiments are completed, an approximation for the learning curve is computed by averaging as follows:

$$\mathrm{E}\, e(k)^2 \approx \hat{E}(k) = \frac{1}{L} \sum_{i=1}^{L} e^{(i)}(k)^2, \quad 1 \leq k \leq N.$$

$\hat{E}(k)$ is referred to as an *ensemble-average* learning curve.

Although it is less common, we can also compute the plot of the MSD versus time. We shall normally refer to this plot also as "the learning curve", or as the *MSD learning curve*, if we need to distinguish between the two plots for MSE and MSD. The MSD ensemble-average learning curve is

$$\mathrm{E}\, \|\tilde{\boldsymbol{w}}_k\|^2 \approx \hat{D}(k) = \frac{1}{L} \sum_{i=1}^{L} \|\tilde{\boldsymbol{w}}_k^{(i)}\|^2, \quad 1 \leq k \leq N.$$

If the step-size μ is sufficiently small, then an average of few tens of experiments is enough to obtain experimental learning curves $\hat{E}(k)$ that are close to the one predicted by independence theory. This one in turn, as mentioned above, approximates the actual learning curve $\mathrm{E}\, e(k)^2$ (*i.e.*, in the absence of the independence assumptions) to first order in μ. Thus ensemble-average learning

curves provide reasonable approximations for the actual learning curve when μ is sufficiently small (we shall also provide an analytical justification for this fact later in Thm. 5.3). But what about larger step-sizes?

First however, let us exemplify the behavior of the ensemble-average learning curve for small step-sizes. Thus consider a length $M = 10$ LMS adaptive filter operating with Gaussian inputs with covariance matrix $\mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T = I$, step-size $\mu = 0.08$, and no noise. The learning curve for this case was computed theoretically in [Rup93], and is given by

$$\mathrm{E}\,e(k)^2 = (1 - 2\mu + 12\mu^2)^k \|\tilde{\boldsymbol{w}}_0\|^2.$$

In Fig. 5.1 we plot this theoretical curve, in addition to an ensemble-average learning curve that is obtained from the average of $L = 100$ simulations. Note how both plots are close to each other, so much so that one is led to believe that there is a single curve in the figure.

Notice, however, how towards the end of the plot both curves start to diverge from each other (see Fig. 5.2). This phenomenon does occur *even* for small step-sizes, and the divergence in fact becomes more pronounced for larger time instants. We shall explain this observation analytically later in Sec. 5.3.5.

Now given the good agreement for sufficiently small step-sizes between the ensemble-average learning curve and the actual learning curve, it is common in the literature to use the average of a few independent repeated experiments to predict or confirm theoretical results from simulation results (a few relatively recent examples include [FW85, VVK94], which use 10-20 independent experiments, and [Rup93, Slo93, TF88], which use 100 independent experiments).

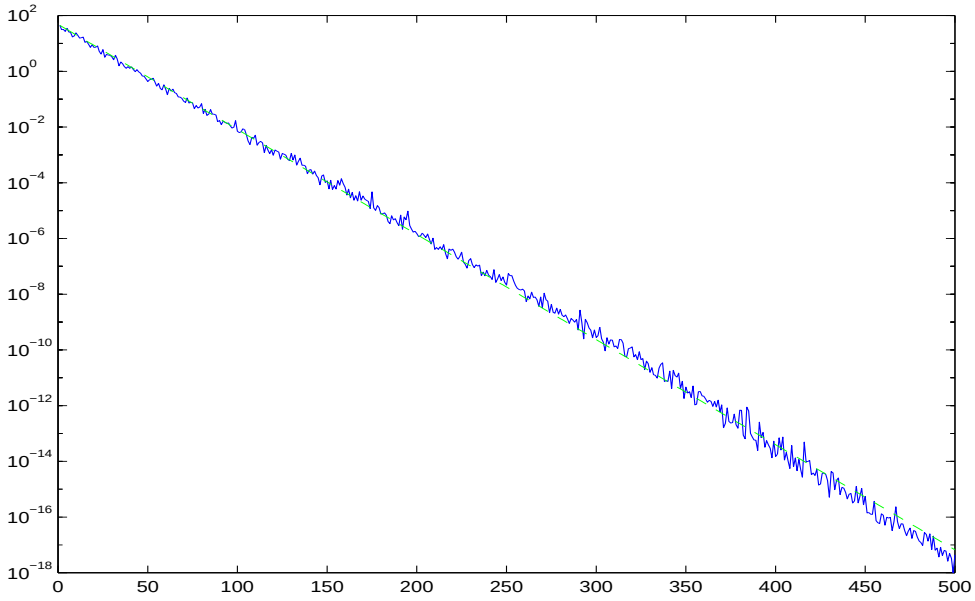

Figure 5.1: *Learning curves computed by simulation and theoretically, with Gaussian iid inputs,* $M = 10$, $\mu = 0.08$, *and* $L = 100$.

5.1.2 Objectives

As mentioned earlier in the introduction, the purpose of this chapter is to show, by examples and also analytically that for larger step-sizes, it may be necessary to perform a considerably larger number of experiments to correctly approximate the average $\mathrm{E}\, e(k)^2$. In other words, we claim that more care is needed while interpreting ensemble-average learning curves. These curves can lead to erroneous conclusions unless a large enough number of experiments are averaged (at times of the order of tens of thousands or higher). We study this phenomenon and provide a theoretical justification for its occurrence. In particular, we establish the six facts listed in the introduction.

In the next section, we provide a few examples that justify the above claims. In Sec. 5.3 we concentrate on the scalar LMS algorithm (*i.e.*, we restrict ourselves

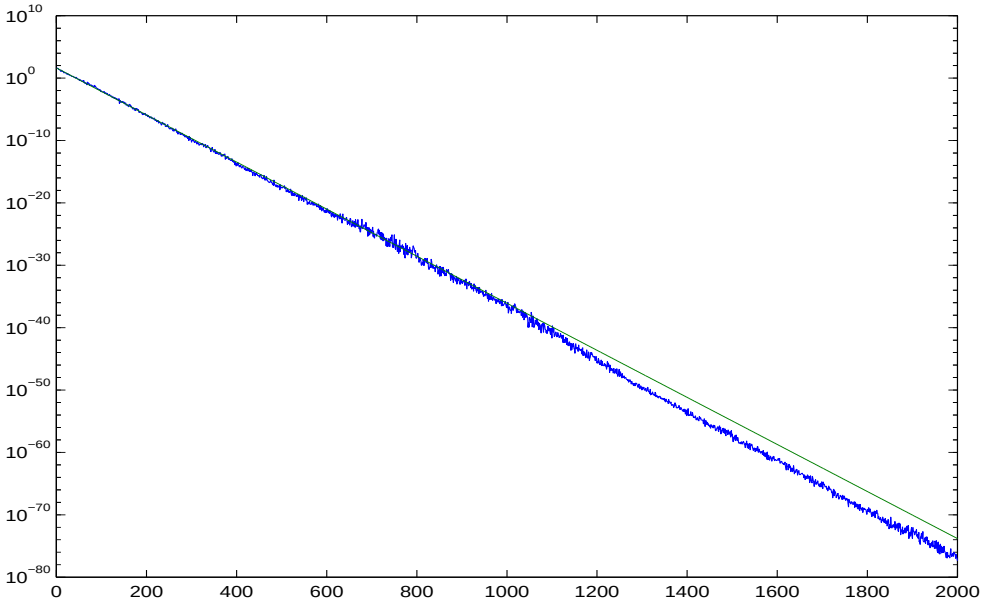

Figure 5.2: *Learning curves computed by simulation and theoretically, with Gaussian iid inputs,* $M = 10$*,* $\mu = 0.08$*, and* $L = 100$*.*

to $M = 1$) with iid inputs and no noise. We show, analytically, that the scalar LMS algorithm has *two* different rates of convergence for larger step-sizes. One rate of convergence is obtained from standard mean-square (MS) analysis, and describes well the behavior of the algorithm for the initial convergence period (k small). The second rate of convergence, which we define and evaluate for several different input distributions, becomes important for larger k. We prove that this second rate of convergence is always faster than that obtained from MS analysis, and that both rates are approximately the same when $\mu \approx 0$. We then show that similar results still hold in the vector case ($M > 1$) in Sec. 5.4.

5.2 SIMULATIONS AND MOTIVATION

The purpose of this section is to demonstrate by means of several simulations that for larger step-sizes there exists a noticeable difference between learning curves derived from MSE analysis and ensemble-average learning curves, even when the latter are constructed by averaging over a large number of repeated experiments. Later we shall show that this phenomenon in fact has an analytical justification and that (for noiseless filters) it cannot be completely removed by indefinitely increasing the number of repeated experiments. We shall also show analytically that this phenomenon disappears for infinitesimally small step-sizes.

We start with a few examples. Consider again the adaptive LMS filter of length $M = 10$ that was used to generate Fig. 5.1, with Gaussian input (i.e., the entries of $\boldsymbol{x}_n$ are Gaussian distributed, with zero mean and variance 1), and Gaussian noise $v(n)$ with variance $\sigma_v^2 = 10^{-4}$. In this case, since the independence assumptions are satisfied, it is possible to compute the learning curve $\mathrm{E}\, e(k)^2$ exactly, using the recursion (2.9) for $\bar{C}_k = \mathrm{E}\, \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T$,

$$\bar{C}_{k+1} = \bar{C}_k - \mu\big(\bar{C}_k R + R\bar{C}_k\big) + 2\mu^2 R\bar{C}_k R + \mu^2 R\,\mathrm{Tr}\big(\bar{C}_k R\big) + \mu^2\sigma_v^2 R,$$

and the expression

$$\mathrm{E}\, e(k)^2 = \mathrm{Tr}\big(R\bar{C}_k\big).$$

In Fig. 5.3, we plot the resulting theoretical learning curve, $\mathrm{E}\, e(k)^2$, as well as ensemble-average learning curves computed with $L = 10$, $L = 100$, and $L = 10,000$, all with step-size $\mu = 0.16$ (which is twice the value of the step-size used to generate Fig. 5.1). Note how all simulation curves are now noticeably far (and, most of the time, below) the (lighter) theoretical curve, although the simulations get closer to the theoretical curve as L is increased. Note also

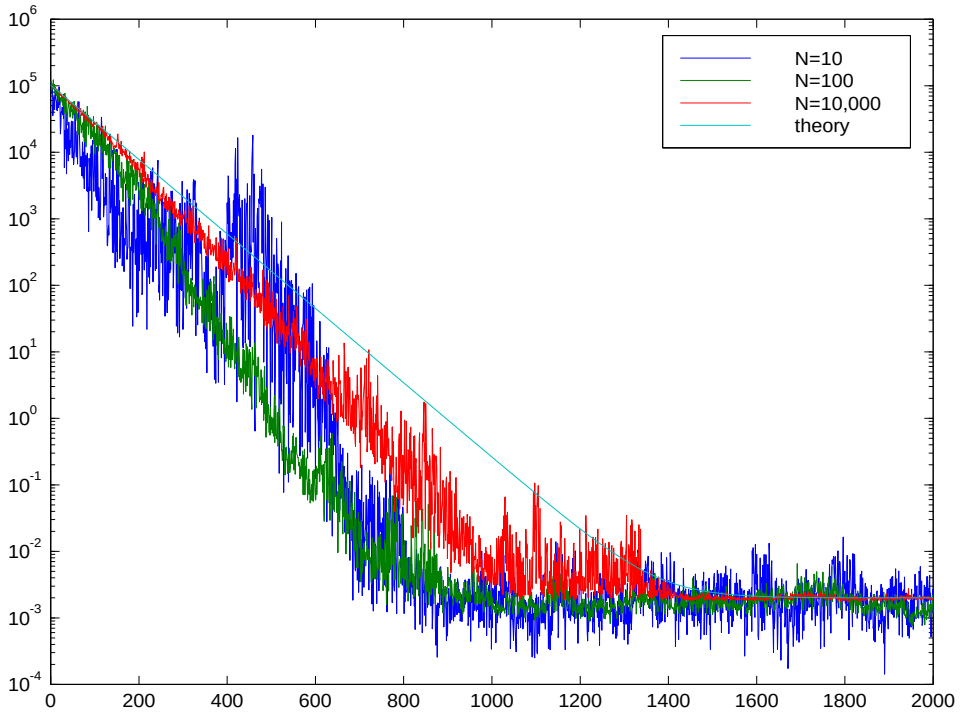

Figure 5.3: *Learning curves computed by simulation and theoretically, with Gaussian independent input vectors, Gaussian noise with* $\sigma_v^2 = 10^{-4}$, $M = 10$, $\mu = 0.16$, *and* $L = 10$, $L = 100$, *and* $L = 10^4$.

that the simulation curves converge faster than the theoretical curve. This situation should be compared with Fig. 5.1, where an almost-perfect agreement was obtained between theory and simulation.

When the independence assumptions do not hold, these effects still occur. In the next example, the input vectors $\boldsymbol{x}_k$ are not iid, but have a delay-line structure, i.e.,

$$\boldsymbol{x}_k = \begin{bmatrix} a(k-M+1) & \dots & a(k-1) & a(k) \end{bmatrix}^T .$$

The results of Chapter 4 can be used to obtain, analytically, the learning curve

$\mathrm{E}\, e(k)^2$ (using the matrix Φ derived for $M = 2$ in Sec. 4.1, Eq. (4.10)). Starting from a deterministic initial condition $\boldsymbol{w}_0$, the vector $\boldsymbol{\Gamma}_0$ (as defined in (4.9)) can be easily computed, since

$$\mathrm{E}\, \tilde{w}_{0,1}^2 = \tilde{w}_{0,1}^2, \qquad \mathrm{E}\big(x_{0,1}\tilde{w}_{0,1}\big)^2 = \tilde{w}_{0,1}^2\, \mathrm{E}\, x_{0,1}^2,$$

and similarly for the other entries of $\boldsymbol{\Gamma}_0$. Having the initial condition $\boldsymbol{\Gamma}_0$ and the state-space matrix Φ, we can compute the MSE using the recursion

$$\boldsymbol{\Gamma}_k = \Phi^k \boldsymbol{\Gamma}_0,$$

and the relation

$$\mathrm{E}\, e(k)^2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \boldsymbol{\Gamma}_k.$$

In Fig. 5.4 we plot this theoretical curve, as well as ensemble-average curves for $L = 100$ to $L = 10,000$, with filter length $M = 2$, step-size $\mu = 8.3$ and for $a(k)$ iid, uniformly distributed between -0.5 and 0.5. With this value of μ, the actual learning curve $\mathrm{E}\, e(k)^2$ can be shown to diverge (and we observe in the figure that it indeed diverges). However, the simulations show the ensemble-average curves $\hat{E}(k)$ converging (see Fig. 5.4 (a)) for various values of L! Notice however that for increasing L, the ensemble-average curve stays closer to the actual learning curve for a longer period of time towards the beginning of the simulation — still the curves separate afterwards with the actual curve diverging and the ensemble-average curve converging (no matter how large L is). We shall explain this fact analytically in Sec. 5.3.6.

These simulations show that the behavior of the ensemble-average curves may be significantly different than that of the theoretical learning curves (*e.g.*, convergence can occur even when divergence by MSE analysis is expected; faster

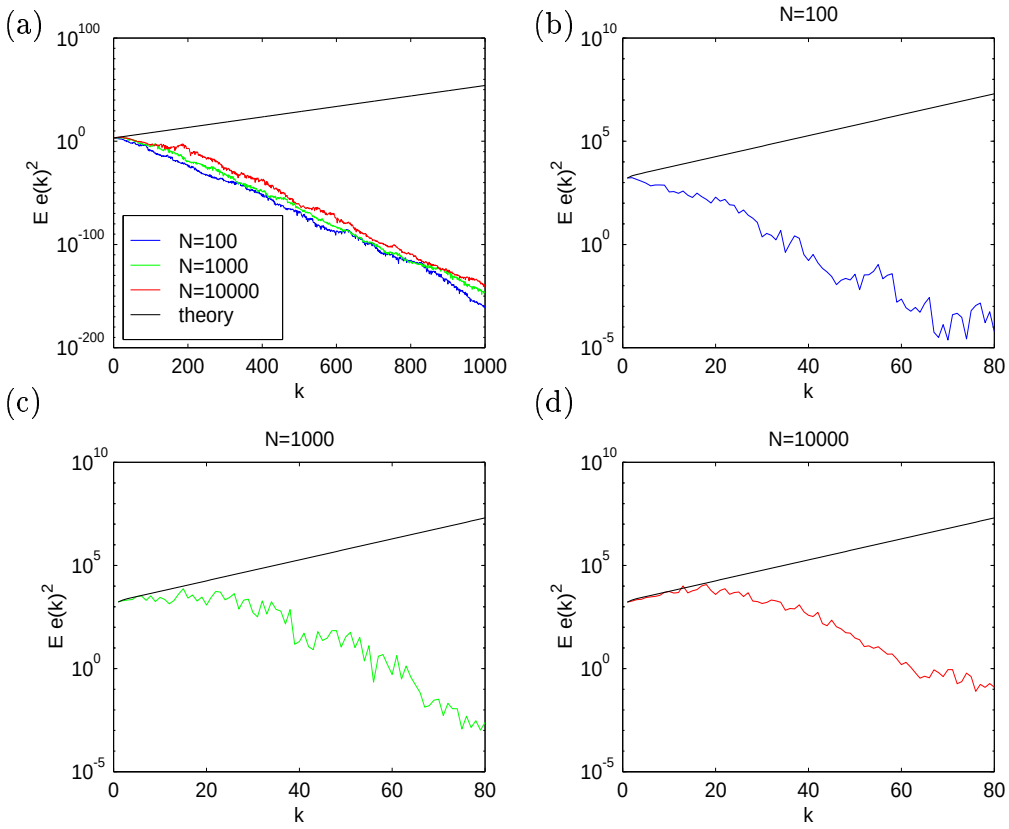

Figure 5.4: *Learning curves computed by simulation and theoretically, with tap-delayed input vectors,* $M = 2$, $\mu = 8.3$, *and* $L = 100$, $L = 1,000$, *and* $L = 10,000$ *(a); theoretical curve and* $L = 100$ *only (b); theoretical and* $L = 1000$ *only (c); theoretical and* $L = 10,000$ *only (d).*

convergence can occur even when slower rates are predicted by MSE analysis, and so on). These differences can lead to wrong conclusions when one attempts to predict performance from simulation results. The interesting fact to stress is that these differences may occur even for very large L.

In the next sections we explain the origin of these effects by focusing first on the scalar LMS case, which serves as a good demonstration and helps highlight

the main ideas.

5.3 THEORETICAL ANALYSIS IN THE SCALAR CASE

A simple model is used in this section to explain the differences observed between the simulations and theoretical results. More specifically, we study the scalar LMS recursion with the following assumptions

I-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is independent.*

I-2. $y(k)$ *is correlated with* $\boldsymbol{x}_k$*, but is independent of all* $\boldsymbol{x}_j$ *with* $j \neq k$*.*

Z. *The noise is identically zero* ($v(k) \equiv 0$)*.*

Thus assuming $M = 1$, we obtain a single-tap adaptive filter with an update equation of the form

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \boldsymbol{x}_k e(k) \ , \quad y(k) = \boldsymbol{x}_k \boldsymbol{w}_* \ , \quad e(k) = y(k) - \boldsymbol{x}_k \boldsymbol{w}_k \ , \tag{5.1}$$

where all variables $\{\boldsymbol{w}_k, \boldsymbol{x}_k, e(k)\}$ are now scalar-valued. Recall that the weight error vector is denoted by $\tilde{\boldsymbol{w}}_k = \boldsymbol{w}_* - \boldsymbol{w}_k$, which therefore satisfies the recursion

$$\tilde{\boldsymbol{w}}_{k+1} = (1 - \mu \boldsymbol{x}_k^2)\tilde{\boldsymbol{w}}_k, \qquad \text{with initial condition } \tilde{\boldsymbol{w}}_0.$$

5.3.1 Condition for Mean-Square Stability

We first determine conditions on the step-size μ for the above one-tap filter to be mean-square stable, *i.e.,* for the variance of $\tilde{\boldsymbol{w}}_k$, $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^2$, to converge to zero. This is a standard step in mean-square stability analysis.

We start by squaring both sides of the above LMS error equation to obtain

$$\tilde{\boldsymbol{w}}_{k+1}^2 = \left(1 - \mu \boldsymbol{x}_k^2\right)^2 \tilde{\boldsymbol{w}}_k^2 \ . \tag{5.2}$$

This is a stochastic difference equation relating two positive quantities, $\tilde{\boldsymbol{w}}_{k+1}^2$ and $\tilde{\boldsymbol{w}}_k^2$. The relation between both quantities is a random multiplicative factor, which we shall denote by

$$u(k) \stackrel{\Delta}{=} \left(1 - \mu \boldsymbol{x}_k^2\right)^2.$$

Note that from our assumptions on $\{\boldsymbol{x}_k\}$, it follows that the $\{u(k)\}$ are iid. To simplify the notation further, we also denote

$$Y_k \stackrel{\Delta}{=} \tilde{\boldsymbol{w}}_k^2 \, .$$

In our simplified notation, the recursion (5.2) becomes

$$Y_{k+1} = \; u(k) Y_k \; = \; Y_0 u(1) u(2) \ldots u(k) \, , \tag{5.3}$$

where the initial condition, $Y_0 = \tilde{\boldsymbol{w}}_0^2$, is assumed *deterministic*.

As mentioned above, we want to determine conditions under which $\mathrm{E}\, Y_k$ converges to zero. For this purpose, we denote the variance and the fourth-order moment of the regressor $\boldsymbol{x}_k$ by

$$\sigma_2 \stackrel{\Delta}{=} \mathrm{E}\, \boldsymbol{x}_k^2 \, , \qquad\qquad \sigma_4 \stackrel{\Delta}{=} \mathrm{E}\, \boldsymbol{x}_k^4 \, .$$

From (5.3), and using the independence of the $\{u(i)\}$, we then obtain

$$\mathrm{E}\, Y_{k+1} = \left(\mathrm{E}\, u\right)^{k+1} Y_0 = \left[1 - \mu\sigma_2 + \mu^2 \sigma_4\right]^{k+1} Y_0 \, , \tag{5.4}$$

where we are denoting the iid nonnegative variables $u(i)$ generically by u (their expected value is equal to $1 - \mu\sigma_2 + \mu^2\sigma_4$).

From the above equation, we conclude that $\mathrm{E}\, Y_k$ will converge to 0 if, and only if, μ is such that the mean of u is strictly less than 1, which leads to

$$0 < \mu\sigma_4/\sigma_2 < 2 \, . \tag{5.5}$$

We will refer to the resulting rate of convergence of $\mathrm{E}\, Y_k$, *viz.*, $\mathrm{E}\, u = 1 - 2\mu\sigma_2 + \mu^2\sigma_4$, as the *MS rate of convergence.*

For ease of comparison with a later condition (see (5.10) further ahead), we shall rewrite the requirement $\mathrm{E}\, u < 1$ in the equivalent form

$$\ln(\mathrm{E}\, u) < 0\ . \tag{5.6}$$

Observe further that the logarithm of the MS rate of convergence is equal to $\ln\big(\mathrm{E}\, u\big)$ (a result that we shall also invoke later – in the discussion following (5.10)). We summarize the above conclusions in the following statement.

Theorem 5.1 (Mean-square stability). *Consider the scalar LMS algorithm (5.1) with stationary iid inputs* $\{\boldsymbol{x}_k\}$. *Assume also that the noise is identically zero. Then,* $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$ *tends to zero if, and only if, the step-size* μ *is such that* $\ln(\mathrm{E}\, u) < 0$ *or, equivalently, (5.5) holds.*

$\diamondsuit$

What we just did was the standard mean-square stability analysis using independence theory.

Consider now the square error $e(k)^2 = \boldsymbol{x}_k^2 \tilde{\boldsymbol{w}}_k^2$. The behavior of $\mathrm{E}\, e(k)^2$ is the same as that of $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$, since $\boldsymbol{x}_k^2$ is stationary and independent of $\tilde{\boldsymbol{w}}_k^2$ and

$$\mathrm{E}\, e(k)^2 = \sigma_2\ \mathrm{E}\, \tilde{\boldsymbol{w}}_k^2\ .$$

Therefore, $\mathrm{E}\, e(k)^2$ will converge when $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$ does, and the rates of convergence will be the same.

5.3.2 Behavior of a Sample Curve

Our experiments in Sec. 5.2 showed that there is a clear distinction between the plots of $\mathrm{E}\,Y_k$, and of the ensemble-average curve $\frac{1}{L}\sum_{i=1}^{L} Y_k^{(i)}$. We explain this fact in this section.

For this purpose, we now focus on the behavior of a typical (single) curve Y_k and show that, for large k, Y_k decays (or increases) at a rate significantly different than that of $\mathrm{E}\,Y_k$. We obtain this result by studying conditions under which a typical curve Y_k converges to zero with probability one (or almost surely). Later, we shall show how this effect manifests itself when several such curves are averaged together to yield an ensemble-average curve.

We start by computing the logarithm of Y_k in Eq. (5.3),

$$\ln Y_k = \ln Y_0 + \sum_{i=0}^{k-1} \ln u(i) \;,$$

which shows that the difference $\ln Y_k - \ln Y_0$ is equal to the sum of k independent and identically distributed random variables ($\ln u(i)$). We assume for now that the variance of $\ln u(i)$ is bounded (Thm. 5.3 gives conditions for this to hold). Therefore, we can use the strong law of large numbers [Dur96] to conclude that

$$\frac{\ln Y_k}{k} \quad \xrightarrow{a.s.} \quad \mathrm{E}\big(\ln u(i)\big) \;\triangleq\; \mathrm{E}\big(\ln u\big) \;, \tag{5.7}$$

where a.s. denotes almost-sure convergence. That is, for large k, $(\ln Y_k)/k$ will almost surely converge to a constant, $\mathrm{E}\ln u$. [We shall evaluate $\mathrm{E}(\ln u)$ for different input distributions in Sec. 5.3.4.]

We now need to translate the above result directly in terms of Y_k, instead of its logarithm. To do so, we must find how fast is the convergence of $\big(\ln Y_k\big)/k$ to its limit. We use a result from [Dur96, pp. 66 and 437] stating that, with

probability one, it holds that

$$\limsup_{k\to\infty}\left(\frac{\ln Y_k - \ln Y_0 - k\,\mathrm{E}\big(\ln u\big)}{k^{1/2}\big(\ln\ln k\big)^{1/2}}\right) = \sqrt{2}\;\sigma_{\ln u} \tag{5.8}$$

where $\sigma^2_{\ln u}$ denotes the variance of $\ln u$, which we assume to be finite (see Theorem 5.3).

Relation (5.8) can be interpreted as follows. Denote by ω the experiment of choosing a regressor sequence $\{\boldsymbol{x}_k\}_{k=1}^{\infty}$. For each experiment ω, compute the resulting sequence $Y_k(\omega)$ for all $k \geq 1$ (starting always from the same initial condition Y_0). Then the statement (5.7) that "$(\ln Y_k)/k$ converges to $\mathrm{E}\ln\big(u\big)$ a.s." means that the set of experiments

$$\mathcal{Z} = \left\{\omega \;\text{ such that }\; \frac{\ln Y_k(\omega)}{k} \to \mathrm{E}\ln\big(u\big)\right\}$$

has probability 1. Moreover, equation (5.8) implies that, with probability one, there exists for each experiment a finite positive number $K(\omega)$ (dependent on the experiment) such that for all $k \geq K(\omega)$, the corresponding curve $Y_k(\omega)$ satisfies

$$\ln Y_k(\omega) = k\,\mathrm{E}\big(\ln u\big) + \ln Y_0 + \delta(k),$$

where the error $\delta(k)$ satisfies

$$|\delta(k)| \leq \sqrt{2}\;\sigma_{\ln u} k^{1/2}\big(\ln\ln k\big)^{1/2}.$$

We stress that $K(\omega)$ depends on the experiment ω.

Therefore, (5.7) and (5.8) imply that, with probability one, a typical curve $(k, Y_k(\omega))$ will eventually enter and stay inside the set

$$\Theta = \left\{(k,\, y(k)) :\; y(k) \leq Y_0 e^{k\,\mathrm{E}\ln u} e^{\sqrt{2k\ln(\ln k)}\sigma_{\ln u}}\right\}. \tag{5.9}$$

In other words, for k large enough, a typical curve $Y_k(\omega)$ will be upper bounded by the curve $Y_0 e^{k\,\mathrm{E}\ln u} e^{\sqrt{2k\ln(\ln k)}\sigma_{\ln u}}$. The convergence of $Y_k(\omega)$ to the above set,

however, is not uniform. That is, there is *no* finite K_0 such that for almost all experiments, $\big(k,\, Y_k(\omega)\big) \in \Theta$ for $k \geq K_0$.

Now since $\mathrm{E}\ln u$ does not depend on the time k, the first exponential in (5.9) dominates the second when k is large, which implies that the upper bound

$$f(k) \stackrel{\Delta}{=} Y_0 e^{k\,\mathrm{E}\ln u} e^{\sigma_{\ln u}\sqrt{2k\ln(\ln k)}} \to 0$$

if, and only if, $\mathrm{E}(\ln u) < 0$. We thus conclude that a typical curve Y_k converges to zero almost surely (or with probability one) if, and only if, the step-size μ is such that

$$\mathrm{E}\ln u < 0 \tag{5.10}$$

This leads to a different condition on μ than the one derived for mean-square stability in (5.6). Note also that for large k, when $\big(k, Y_k\big)$ is already close to or inside Θ, the rate of convergence of a typical curve Y_k is dictated primarily by the term $e^{k\,\mathrm{E}\ln u}$. This implies that, for large k, the logarithm of the rate of convergence of $Y_k = \tilde{\boldsymbol{w}}_k^2$ is given by $\mathrm{E}\ln u$ (which should be contrasted with $\ln \mathrm{E}\, u$ in the mean-square analysis case right after (5.6)). We shall refer to this second rate as the *a.s. rate of convergence*. The following theorem has thus been proved.

Theorem 5.2 (Almost-sure convergence). *Consider the scalar LMS algorithm (5.1) with stationary iid inputs $\{\boldsymbol{x}_k\}$. Assume also that the noise is identically zero. Then, with probability one, there is a finite constant K (dependent on the realization) such that $\big(k,\, \tilde{\boldsymbol{w}}_k^2\big)$ stays inside the set Θ defined above for all $k \geq K$. In particular, a typical curve $\tilde{\boldsymbol{w}}_k^2$ converges to zero with probability one if, and only if, $\underline{\mathrm{E}\ln u < 0}$ (which is equivalent to $\mathrm{E}\ln\big(1-\mu\boldsymbol{x}_k^2\big)^2 < 0$).*

$\diamondsuit$

There are related works in the literature that have also studied the almost-sure stability of LMS (e.g., [BA81, BAN86, Sol97]), or even of continuous-time systems (e.g., [Koz69, PE78]). None of these works, however, highlights the distinctions that arise between mean-square stability and almost-sure stability in the same level of generality that we do here. Reference [BAN86], for example, compares both notions of stability for $\mu \approx 0$, when they in fact agree, proving a version of Theorem 5.3 further ahead. There are no prior results on how, more generally, the two notions compare for non-infinitesimal values of μ, and on how these distinctions provide a natural explanation for the fact that LMS has two distinct rates of convergence. There are also no prior results on how ensemble average learning curves are affected by these two notions of stability.

5.3.3 Comparisons

Comparing the statements of Thms 5.1 and 5.2 we see that there is a fundamental difference in the conditions required for convergence in both cases. The first theorem shows that mean-square convergence requires the step-size μ to be such that $\ln \mathrm{E}\, u < 0$, while the second theorem shows that almost-sure convergence requires μ to be such that $\mathrm{E} \ln u < 0$. The two conditions are not equivalent and, in fact, one implies the other since, for any nonnegative random variable u for which $\mathrm{E}\, u$ and $\mathrm{E} \ln u$ both exist, it holds that

$$\mathrm{E}\big(\ln u\big) \leq \ln\big(\mathrm{E}\, u\big).$$

This result follows directly from Jensen's inequality since the function $\big(-\ln x\big)$ is convex (*e.g.,* [Dur96, p. 14] and [CT91]). Therefore, values of μ for which almost-sure convergence occurs also guarantee mean-square convergence while the converse is not true. A value for which $\ln \mathrm{E}\, u > 0$ (and thus mean-square divergence occurs) can still guarantee almost-sure convergence, or $\mathrm{E} \ln u < 0$.

We shall elaborate more on these distinctions in the sequel and explain how they can be used to explain the phenomena that we observed in the simulations in Sec. 5.2. For now, however, we show that these distinctions disappear for infinitesimally small step-sizes (a fact that does not depend on specific input signal distribution). The full proof is a bit elaborate and requires several auxiliary results that are given in Appendix 5.A. Here we summarize the main conclusion. [We remark that the requirement on the probability density function in the statement of the theorem is not restrictive and it does not rule out most well-known distributions.]

Theorem 5.3 (Rate of convergence for small step-sizes). *Let $p(x)$ denote the probability density function of the iid regressor sequence $\boldsymbol{x}_n$. Assume there exist constants $B < \infty$ and $\beta > 5$ such that $p(x)$ satisfies*

$$p(x) \leq \frac{1}{x^\beta} \quad \textit{for } |x| \geq B.$$

Then the quantities $\mathrm{E}\ln\big(1-\mu\boldsymbol{x}_k^2\big)^2$, $\mathrm{var}\left(\ln\big(1-\mu\boldsymbol{x}_k^2\big)^2\right)$, *and* $\ln\mathrm{E}\big(1-\mu\boldsymbol{x}_k^2\big)^2$ *exist, are finite, and satisfy*

$$\mathrm{E}\ln\big(1-\mu\boldsymbol{x}_k^2\big)^2 = -2\mu\,\mathrm{E}\,\boldsymbol{x}_k^2 + o(\mu),$$
$$\ln\mathrm{E}\big(1-\mu\boldsymbol{x}_k^2\big)^2 = -2\mu\,\mathrm{E}\,\boldsymbol{x}_k^2 + o(\mu),$$

where $o(\mu)$ is a function satisfying

$$\lim_{\mu\to 0}\frac{o(\mu)}{\mu} = 0.$$

$\diamondsuit$

The theorem therefore shows that $\mathrm{E}\ln u$ and $\ln\mathrm{E}\,u$ are approximately the

same when μ is infinitesimally small since it implies that

$$\mathrm{E}\ln u = -2\mu\sigma_2 + o(\mu)\ , \quad \ln \mathrm{E}\, u = -2\mu\sigma_2 + o(\mu)\ .$$

This explains why learning curves and ensemble-average learning curves tend to agree reasonably well for such small step-sizes.

5.3.4 Some Examples

We now provide a few examples showing that for larger step-sizes, the difference between $\mathrm{E}\ln u$ and $\ln \mathrm{E}\, u$ can be considerably large. In particular, the difference can be large around the step-size that achieves fastest convergence.

Uniform input sequence

Here we assume that $\boldsymbol{x}_k$ is a uniform random variable, with values in the interval $[-\alpha,\ \alpha]$. For this input distribution we have $\sigma_2 = \alpha^2/3$ and $\sigma_4 = \alpha^4/5$, so that

$$\ln \mathrm{E}\, u \ = \ \ln \mathrm{E}\big(1 - \mu \boldsymbol{x}_n^2\big)^2 = \ln\left(1 - 2\mu\frac{\alpha^2}{3} + \mu^2\frac{\alpha^4}{5}\right).$$

We can also evaluate $\mathrm{E}\ln u$ as a function of $\mu\alpha^2$ theoretically and obtain (the formulas below were obtained using MAPLE [Red93] to compute the integral $\int_{-\alpha}^{\alpha} \frac{1}{\alpha}\ln x \,\mathrm{d}\, x$)

$$\mathrm{E}\ln u \ = \ \begin{cases} \ln\big(1-\mu\alpha^2\big)^2 + \frac{4}{\alpha\sqrt{\mu}}\mathrm{arctanh}\big(\alpha\sqrt{\mu}\big) - 4, & \text{if} \quad \mu\alpha^2 \le 1, \\ \\ \ln\big(1-\mu\alpha^2\big)^2 + \frac{4}{\alpha\sqrt{\mu}}\mathrm{arccoth}\big(\alpha\sqrt{\mu}\big) - 4, & \text{if} \quad \mu\alpha^2 > 1. \end{cases}$$

Fig. 5.5 compares the plots of $\mathrm{E}\ln u$ (the continuous line in the figure) and of $\ln\big(\mathrm{E}\, u\big)$. Note that both plots are close together for small $\mu\alpha^2$ (as predicted by Thm. 5.3), but they become significantly different as $\mu\alpha^2$ increases. In particular,

they are quite different at the minima of each plot (which correspond to the fastest rates of convergence from the MS and almost-sure convergence points of view). In the ranges of $\mu\alpha^2$ for which the curves are significantly different, the rate of convergence of a typical curve Y_k will be significantly different than the rate of convergence of $\mathrm{E}\, Y_k$ (for large k).

With this result we can explain why the ensemble-average curves computed for small step-sizes are close to the "theoretical" predictions using $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$, and why these plots are so different for larger step-sizes. For sufficiently small step-sizes, the rates of convergence of both $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$ and of $\tilde{\boldsymbol{w}}_k^2$ are, with probability one, very close, so we expect that an average of a few simulations will produce a reasonable approximation for $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$. For larger step-sizes, and for large k, however, the rate of convergence of $\tilde{\boldsymbol{w}}_k^2$ is significantly different (and faster) than that predicted by Eq. (5.4). Thus we should expect to need a larger number of simulations to obtain a good approximation for $\mathrm{E}\, \tilde{\boldsymbol{w}}_k^2$. This latter point will be better clarified by the variance analysis that we provide in Sec. 5.3.6.

Another interesting observation is that $\mathrm{E} \ln u$ is negative well beyond the point where $\ln\big(\mathrm{E}\, u\big)$ becomes positive. This implies that there is a range of step-sizes for which a typical curve Y_k converges to zero with probability one, but $\mathrm{E}\, Y_k$ diverges. This explains the simulations in Fig. 5.4. This is *not* a paradox. Since the convergence is not uniform, there is a small (but nonzero) probability that a sample curve Y_k will exist such that it assumes very large values for a long interval of time before converging to zero.

For an example of these affirmations, refer to Fig. 5.6, in which we used $\mu\alpha^2 = 5.5$, a choice of step-size for which $\mathrm{E}\, Y_k$ diverges with rate 3.383^n. On the other hand, our results show that a sample curve Y_k should converge almost-surely at the rate 0.8605^k. In the figure we plot 2 simulations of Y_k (without

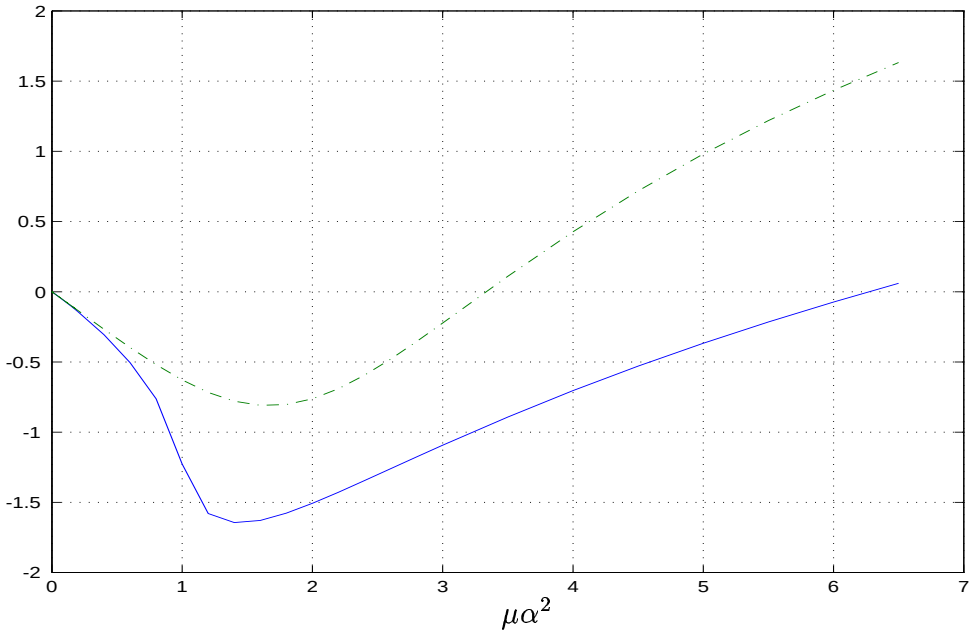

Figure 5.5: *Graphs of* $\mathrm{E}\ln\left(1-\mu\boldsymbol{x}_k^2\right)^2$ *(continuous line) and* $\ln\mathrm{E}\left(1-\mu\boldsymbol{x}_k^2\right)^2$ *(broken line).*

averaging), in addition to the plot of $0.8605^k Y_0$. Although the two simulations start growing, both plots eventually start to converge with the rate predicted by our almost sure analysis.

Gaussian inputs

Assume now that $\boldsymbol{x}_k$ is Gaussian with zero mean and unit variance, so that $\sigma_2 = 1$ and $\sigma_4 = 3$. Then

$$\mathrm{E}\, Y_k = \left(1 - 2\mu + 3\mu^2\right)^k Y_k.$$

We computed $\mathrm{E}\ln\left(1-\mu\boldsymbol{x}_k^2\right)^2$ numerically (using Maple from the symbolic toolbox in Matlab [Red93]), obtaining the results shown in Fig. 5.7. Note again, how $\ln\mathrm{E}\, u$ and $\mathrm{E}\ln u$ are approximately equal for small μ. An interesting fact that appears here is that the value of μ that achieves fastest MS convergence is significantly smaller than the step-size that achieves fastest almost-sure convergence.

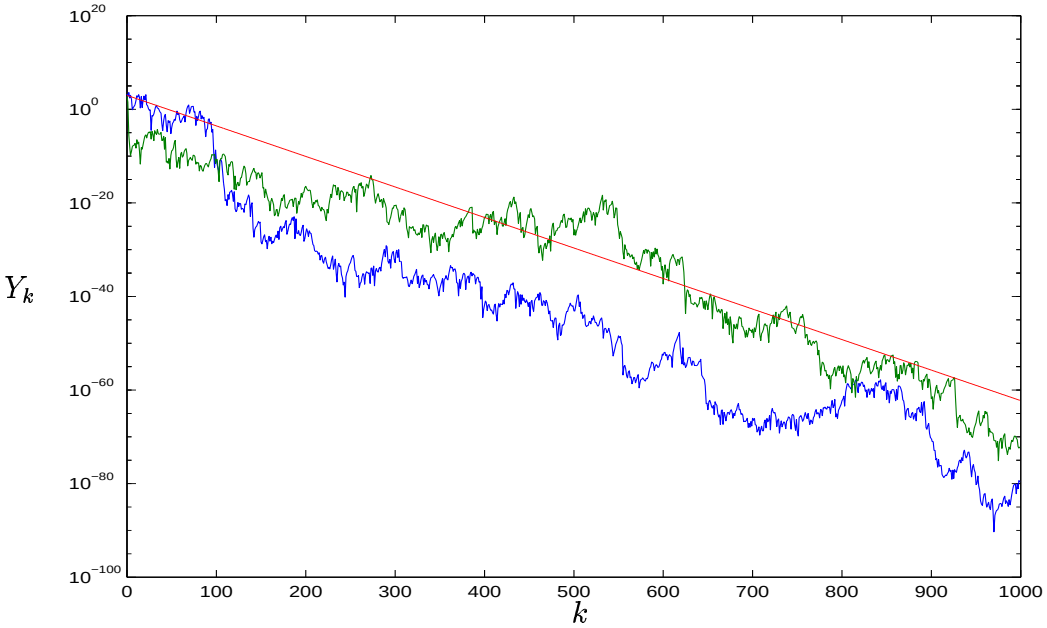

Figure 5.6: *Two plots of Y_k (not averaged) for $\mu\alpha^2 = 5.5$. Also plotted is the almost-sure rate of convergence* $0.8605^k Y_0$.

In distributions with heavier tails (such as the exponential), this difference is even more pronounced.

5.3.5 Differences Between Theory and Simulation

The above results can thus be used to understand the differences between theoretical and simulated learning curves for large k and for larger step-sizes. Indeed, let $\{\omega_l\}_{l=1}^{L}$ be L independent experiments, with the corresponding sample curves $\{Y_k(\omega_l)\}$. We know from Thm. 5.2 that for each curve there exists an integer $K(\omega_l)$ such that $Y_k(\omega_l)$ will remain inside the set Θ for all $k \geq K(\omega_l)$. In particular, if the step-size is such that $\mathrm{E}\ln\left(1-\mu\boldsymbol{x}_k^2\right)^2 < 0$, then this means that, with probability one, $Y_k(\omega_l)$ will be converging to zero for all $k \geq K(\omega_l)$.

Now let $\hat{Y}_k = \frac{1}{L}\sum_{l=1}^{L} Y_k(\omega_l)$ be the ensemble-averaged learning curve. Since

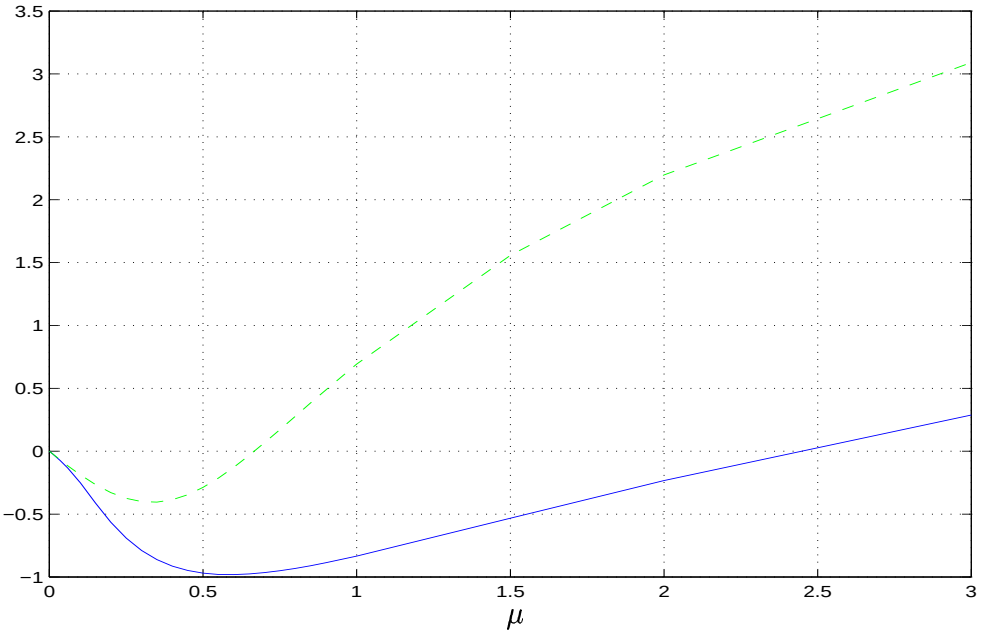

Figure 5.7: $\mathrm{E}\ln u$ *(continuous line) and* $\ln \mathrm{E}\, u$ *(broken line) for Gaussian input with unit variance.*

$\big(k, Y_k(\omega_l)\big)$ stays inside Θ for $n \geq K(\omega_l)$, $\big(k, \hat{Y}_k\big)$ will also stay inside Θ for $k \geq \bar{K} = \sup K(\omega_l)$. This means that eventually (for large enough k), all ensemble-averaged learning curves will stay far away from the average curve $\mathrm{E}\, Y_k$. This is because the actual average curve $\mathrm{E}\, Y_k$ and typical sample curves Y_k will converge at different rates for large k (one rate of convergence is dictated by $\ln \mathrm{E}\, u$ while the other is dictated by $\mathrm{E}\ln u$).

Thus we can say that the almost-sure analysis allows us to clarify what happens when we fix L (the number of repeated experiments) and increase k (the time variable). Indeed, in this situation, the ensemble-average curve tends to be significantly different from the true average curve for increasing k due to the difference in the convergence rates.

On the other hand, the more simulations we average, the larger we expect $\bar{K}$

to be, so that the difference between the ensemble-average curve and the true average will be significant only for increasingly large k. That is, the more we average, the longer it takes for us to see the difference between the ensemble-average curve and the true-average curve. We shall explain this fact more clearly in Sec. 5.3.6 by means of a variance analysis. We summarize these conclusions in the following statement for ease of reference.

Theorem 5.4 (Almost-sure analysis). *Consider the scalar LMS algorithm (5.1) with stationary iid inputs* $\{\boldsymbol{x}_k\}$. *Assume that the noise is identically zero and that the distribution of* u *is such that* $\mathrm{E}\ln u < \ln \mathrm{E}\, u$ *(i.e., strict inequality holds). Then the following conclusions hold:*

1. *If we fix* L, *then for large enough* k, *the ensemble-average curve will be noticeably different from the true-average curve due to different rates of convergence.*

2. *The more we average (i.e., the larger the value of* L*), the longer it takes for the difference between the ensemble-average curve and the true-average curve to be noticed.*

$\diamondsuit$

5.3.6 Variance Analysis

The almost-sure convergence analysis of the previous sections establishes that for large enough time k, there always exists a difference between the ensemble-average curve and the true-average curve, and that this difference is explained by the fact that the convergence rates of both curves are distinct. In view of this, we

shall say that the almost-sure analysis helps us explain the distinction between both curves for large time instants k.

If we however reexamine the curves of Figs. 5.3 and 5.4, we see that for small k (that is, close to the beginning of the curves), there is usually a good match between the learning curve and the ensemble-average curve. Put in another way, we notice that the rates of convergence of the true learning curve and the ensemble-average curve tend to be identical for initial time instants. Only for later time instants, the rates of convergence become different as predicted by the almost-sure analysis.

To explain this initial effect, we rely on a different argument that employs the Chebyshev's inequality. It will allow us to clarify what happens for smaller time instants. In particular, the analysis will show that a sample curve Y_k "stays close" to the average $\mathrm{E}\, Y_k$ for small k (but not necessarily so for large k, as argued before). How small k must be so that Y_k is close to $\mathrm{E}\, Y_k$ depends on the value of the step-size and on the distribution of $\boldsymbol{x}_k$. The analysis will also allow us to clarify what happens when we fix the time instant k and increase the number of experiments L.

We start by evaluating the variance of Y_k, $\mathrm{var}\, Y_k$, rather than its mean (as in Sec. 5.3.1). This is because we shall study the evolution of the following ratio

$$\rho(k) \stackrel{\Delta}{=} \frac{\sqrt{\mathrm{var}\, Y_k}}{\mathrm{E}\, Y_k}\,, \tag{5.11}$$

which we stress is a function of the time instant k; that is, with each k we associate a value $\rho(k)$. We claim that for values of k for which $\rho(k) \ll 1$, the average value $\mathrm{E}\, Y_k$ will be a good approximation for the values of the sample curve, Y_k, at these time instants. To see this, assume that $\rho(k) = 0.05$ for a particular value of k.

Using Chebyshev's inequality [Dur96, p. 15], we obtain

$$\mathrm{P}\left\{\left|Y_k - \mathrm{E}\,Y_k\right| \geq \frac{1}{2}\,\mathrm{E}\,Y_k\right\} \leq \frac{\rho(k)^2\left(\mathrm{E}\,Y_k\right)^2}{0.25\left(\mathrm{E}\,Y_k\right)^2} = 0.01.$$

This means that we have a 99% probability that Y_n will be in the interval $\left[0.5\,\mathrm{E}\,Y_k,\, 1.5\,\mathrm{E}\,Y_k\right]$.

Now, when we form ensemble-average learning curves, we average several sample curves Y_k to obtain

$$\hat{D}(n) = \frac{1}{L}\sum_{i=1}^{L} Y_k^{(i)}\ .$$

Assuming that we use L independent experiments, then the expected value of $\hat{D}(k)$ is still equal to $\mathrm{E}\,Y_n$. The ratio $\rho(k)$ that is associated with the curves $\{\mathrm{E}\,Y_k, \hat{D}(k)\}$ will thus be given by

$$\rho'(k) = \frac{\sqrt{\mathrm{var}\,\hat{D}(k)}}{\mathrm{E}\,Y_k}\ ,$$

where the variance of $\hat{D}(k)$ is equal to

$$\mathrm{var}\,\hat{D}(k) = \frac{\mathrm{var}\,Y_k}{L}\ .$$

This implies that

$$\rho'(k) = \frac{\sqrt{\mathrm{var}\,\hat{D}(k)}}{\mathrm{E}\,Y_k} = \frac{\rho(k)}{\sqrt{L}}. \tag{5.12}$$

That is, the process of constructing ensemble-average curves reduces the value of $\rho(k)$. Therefore, if we choose L large enough, the ensemble-average learning curve should be a good approximation for $\mathrm{E}\,Y_k$, at those time instants where $\rho'(k)$ is sufficiently small.[1]

[1] Although a small $\rho(k)$ implies that $Y_n \approx \mathrm{E}\,Y_k$ and $\hat{D}(k) \approx \mathrm{E}\,Y_k$, a large $\rho(k)$ *does not imply*

Thus a small $\rho(k)$ is desirable to conclude that Y_k or $\hat{D}(k)$ is close to $\mathrm{E}\,Y_k$. However, it turns out that the ratio $\rho(k)$ increases with k (and thus $\hat{D}(k)$ approximates $\mathrm{E}\,Y_k$ less effectively for larger k, which is consistent with the results of our almost-sure analysis). To see this, we evaluate $\mathrm{E}\,Y_k^2$,

$$\mathrm{E}\,Y_k^2 = \left(\mathrm{E}\,u^2\right)^k Y_0^2 = \left(1 - 4\mu\sigma_2 + 6\mu^2\sigma_4 - 4\mu^3\sigma_6 + \mu^4\sigma_8\right)^k Y_0^2,$$

where $\sigma_6 \triangleq \mathrm{E}\,\boldsymbol{x}_n^6$ and $\sigma_8 \triangleq \mathrm{E}\,\boldsymbol{x}_n^8$ are assumed finite. Define

$$r_4 \triangleq \mathrm{E}\,u^2 = \left(1 - 4\mu\sigma_2 + 6\mu^2\sigma_4 - 4\mu^3\sigma_6 + \mu^4\sigma_8\right) \quad \text{and} \quad r_2 \triangleq \left(1 - 2\mu\sigma_2 + \mu^2\sigma_4\right). \tag{5.13}$$

With these definitions, $\rho(k)$ is given by

$$\rho(k) = \frac{\sqrt{r_4^k - r_2^{2k}}}{r_2^k} = \sqrt{\frac{r_4^k}{r_2^{2k}} - 1}. \tag{5.14}$$

Now since

$$0 \le \mathrm{var}\,Y_k = \mathrm{E}\,Y_k^2 - \left(\mathrm{E}\,Y_k\right)^2 = \left(\mathrm{E}\,u^2\right)^k Y_0^2 - \left(\mathrm{E}\,u\right)^{2k} Y_0^2 = r_4^k Y_0^2 - r_2^{2k} Y_0^2,$$

we conclude that $r_4 \ge r_2^2$ (with equality only if $\boldsymbol{x}_k$ is a constant with probability one). Therefore, except for this trivial case, $\rho(k)$ is strictly increasing, and

$$\lim_{k\to\infty} \rho(k) = \lim_{k\to\infty} \frac{\sqrt{r_4^k - r_2^{2k}}}{r_2^k} = +\infty.$$

We have thus proved the following theorem.

---

that the difference $|Y_k - \mathrm{E}\,Y_k|$ or $|\hat{D}(k) - \mathrm{E}\,Y_k|$ should be large with a significant probability (but it does hint that this may be the case). For example, take a random variable y satisfying

$$y = \begin{cases} 10 & \text{with probability } 10^{-4} \\ 10^{-2} & \text{with probability } 0.9999. \end{cases}$$

In this case $\mathrm{E}\,y^2 = 0.0101$ and $\mathrm{E}\,y = 0.0110$, and thus $\rho = 9.1$. Despite the large value of ρ, $|y - \mathrm{E}\,y| \le 10^{-3} = 0.1\,\mathrm{E}\,y$ in 99.99% of the realizations.

Theorem 5.5 (Variance of Y_k). *Assume r_2 and r_4 defined above are finite, and that the initial condition $\boldsymbol{w}_0$ to the scalar LMS algorithm (5.1) is deterministic. Then the ratio $\rho(k)$ between the standard deviation of Y_k and $\mathrm{E}\,Y_k$ is either 0 for all k, or is strictly increasing with k, and tends to infinity as k tends to infinity.*

$\diamondsuit$

Note that from our assumption that Y_0 is deterministic, we obtain $\rho(0) = 0$. In general (for step-sizes for which Y_k converges in the MS sense), $\rho(k)$ remains small for for some time, which implies (via Chebyshev's inequality) that Y_k is well approximated by $\mathrm{E}\,Y_k$ when k is small. We give below examples of the behavior of $\rho(k)$ for two different input distributions: binary and Gaussian $\boldsymbol{x}_k$.

Binary inputs

We first give a simple example for which $\rho(k) \equiv 0$. Assume that

$$\boldsymbol{x}_k = \begin{cases} +1, & \text{with probability } 0.5, \\ -1, & \text{w.p. } 0.5. \end{cases}$$

Under this condition, $u(k) \equiv (1-\mu)^2$ is a constant, and thus $Y_k \equiv \mathrm{E}\,Y_k$. In this trivial case, we have $\sigma_2 = \sigma_4^{1/2} = \sigma_6^{1/3} = \sigma_8^{1/4}$, and $\rho(k) = 0$ for all k.

Gaussian inputs

Let $\boldsymbol{x}_k$ be Gaussian with zero mean and unit variance. Now we have

$$\sigma_2 = 1, \qquad \sigma_4 = 3, \qquad \sigma_6 = 15, \qquad \text{and} \qquad \sigma_8 = 105.$$

We plot the value of $\rho(k)$ for several values of μ in the range $0 < \mu < 2/3 = 2\sigma_2/\sigma_4$ in Fig. 5.8. Note how $\rho(n)$ grows increasingly fast as μ increases. Note also that the rate of increase of $\rho(k)$ is very small for $\mu \approx 0$. This fact provides another

explanation for the reason why $\hat{D}(k)$ approximates $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2$ reasonably well when the step-size is sufficiently small.

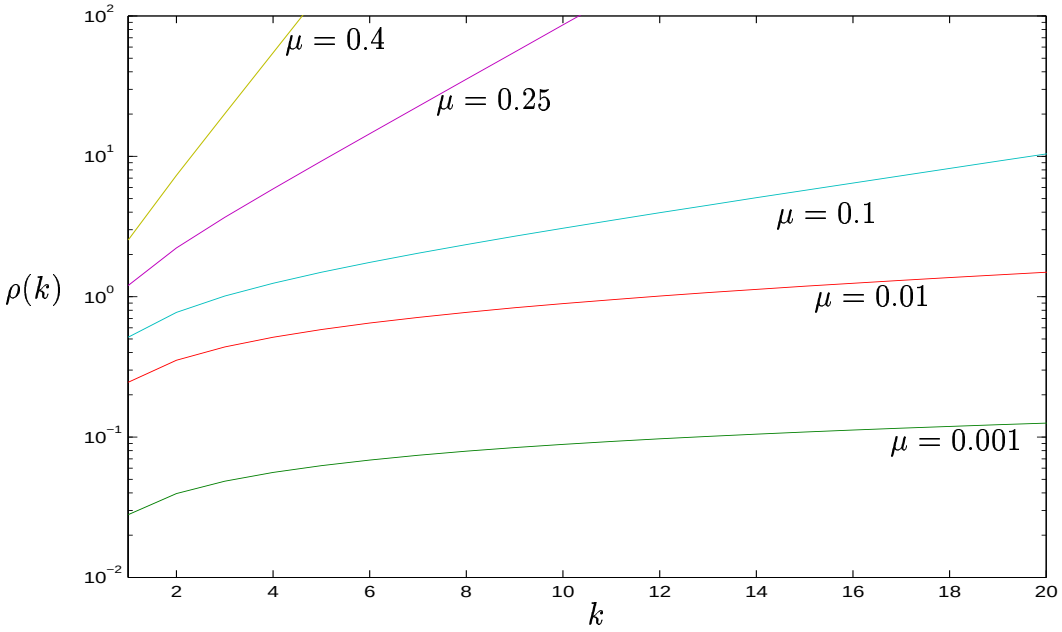

Figure 5.8: $\rho(k)$, *computed for* $\mu = 0.001,\ 0.01,\ 0.1,\ 0.25,$ *and* 0.4.

On the other hand, for larger step-sizes, $\rho(k)$ grows very rapidly. Fig. 5.9 shows $\rho(k)$ for $\mu = 0.33$ (which is slightly smaller than the step-size that gives fastest convergence of $\mathrm{E}\,Y_k$ in this case). We can see from the figure that, if we want to use Chebyshev's bound to guarantee that $\mathrm{P}\left\{\left|Y_{10} - \mathrm{E}\,Y_{10}\right| \geq 0.5\,\mathrm{E}\,Y_{10}\right\} \leq 0.01$, we need to average approximately $L = 34,000$ experiments. For larger values of μ, $\rho(k)$ grows even faster.

Two rates of convergence

Let us consider again the differences between theory and simulation. Assume that we fix the time instant k and compare the values of $\mathrm{E}\,Y_k$ and $\hat{D}(k)$ at that particular time instant. That is, we compare the value of the learning curve with

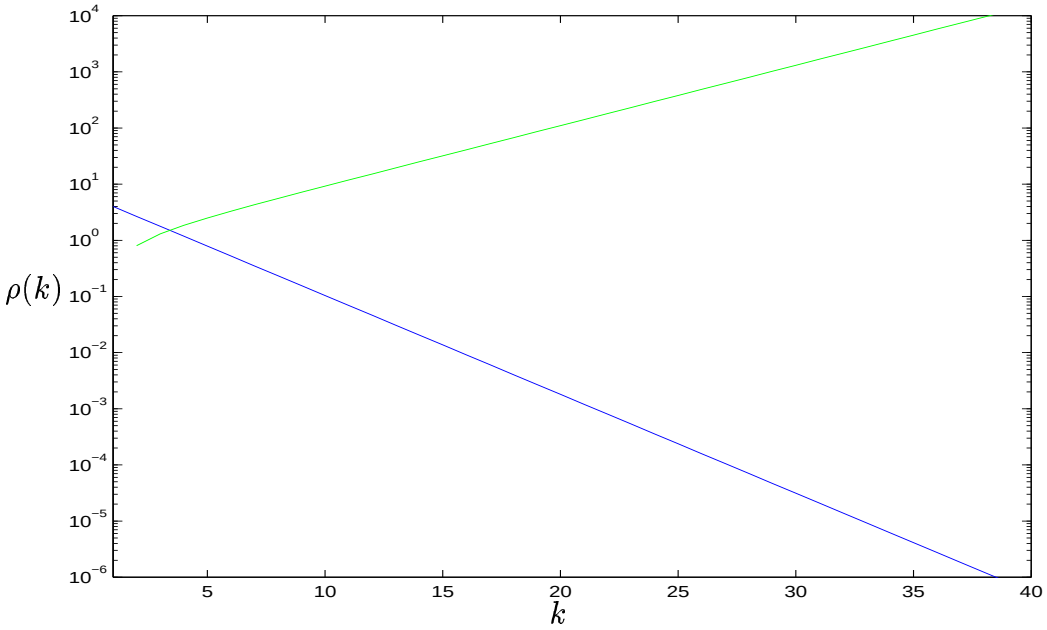

Figure 5.9: $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^2$ *(dark curve) and* $\rho(k)$ *(light curve), computed for* $\mu = 0.33$.

the value of the ensemble-average curve for different values of L.

We know from the expression for $\rho'(k)$ that the larger the value of L, the smaller the value of $\rho'(k)$. Hence, the more we average, the closer will the value of $\hat{D}(k)$ be to EY_k. That is, the closer will be both curves at that time instant k. This again confirms an earlier conclusion in Theorem 5.4, viz., that the more we average, the longer it takes for us to see the differences between both curves.

Another major conclusion that follows from the almost-sure and variance analyses is that the LMS recursion exhibits *two different rates of convergence.* At first, for small k, a sample curve Y_k is close to $\mathrm{E}\,Y_k$ and therefore converges at a rate that is determined by $\mathrm{E}\ln u$. For larger k, the sample curve Y_k will converge at a rate that is determined by $\ln \mathrm{E}\,u$.

5.4 THEORETICAL ANALYSIS IN THE VECTOR CASE

In this section we extend the ideas presented above to larger filter lengths. It turns out that the behavior of the LMS algorithm for filter lengths $M > 1$ is richer than what we saw in the scalar case, and is (except when the step-size is vanishingly small) very dependent on the actual input distribution. Therefore, the examples shown in this section cannot be exhaustive, *i.e.*, the examples do not show all possible kinds of behavior – but they do illustrate the phenomena we are interested in.

As before, we shall provide mean-square, almost-sure, and variance analyses. We start with the latter and explain how to compute the variance of $\|\tilde{\boldsymbol{w}}_k\|^2$ (generalizing the results of Sec. 5.3.6) in the vector case.

5.4.1 Variance Analysis

We continue to assume that the input sequence $\{\boldsymbol{x}_k \in \mathbb{R}^M\}$ is iid and that the noise is identically zero ($v(k) \equiv 0$). The individual entries of each regressor vector $\boldsymbol{x}_k$ however are not assumed to be independent.

In this section we are interested in evaluating the ratio $\rho(k)$ when $M > 1$, and which we define as

$$\rho(k) = \frac{\sqrt{\text{var}\left(\|\tilde{\boldsymbol{w}}_k\|^2\right)}}{E\|\tilde{\boldsymbol{w}}_k\|^2} \,. \tag{5.15}$$

This calculation requires that we evaluate both the mean and the variance of the quantity $\|\tilde{\boldsymbol{w}}_k\|^2$. The value of $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2$ can be evaluated using the Kronecker product method described in Chapter 2. We now concentrate in the evaluation of $\text{var}(\|\tilde{\boldsymbol{w}}_k\|^2)$.

In the following we will often use repeated Kronecker products, as in $A \otimes A \otimes A$.

We shall denote such "Kronecker powers" as $A^{\otimes 2} \triangleq A \otimes A$, and similarly for $k > 2$.

We start the computation of $\mathrm{var}\left(\|\tilde{\boldsymbol{w}}_k\|^2\right)$ by noting that

$$\mathrm{var}\left(\|\tilde{\boldsymbol{w}}_k\|^2\right) = \mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^4 - \left(\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2\right)^2 \ , \tag{5.16}$$

and that, as shown in Appendix 5.B, we can rewrite $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^4$ as

$$\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^4 = \mathrm{E}\,\mathrm{Tr}\left(\left(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\right)^{\otimes 2}\right) \ .$$

The following recursion for $\mathrm{vec}\left(\left(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\right)^{\otimes 2}\right)$ is established in Appendix 5.C.

Theorem 5.6 (Recursion for variance calculation). *The expected value*

$$\mathrm{E}\big(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big)^{\otimes 2}$$

can be computed from the recursion

$$\begin{aligned}\mathrm{E}\,\mathrm{vec}\left(\left(\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T\right)^{\otimes 2}\right) &= \mathrm{E}\left(\left(I-\mu\boldsymbol{x}_k\boldsymbol{x}_k^T\right)^{\otimes 4}\right)\mathrm{E}\,\mathrm{vec}\left(\left(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\right)^{\otimes 2}\right) \\ &\triangleq \Psi\,\mathrm{E}\,\mathrm{vec}\left(\left(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\right)^{\otimes 2}\right).\end{aligned} \tag{5.17}$$

$\diamondsuit$

The above recursion allows us to evaluate $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^4$, which in turn can be used in (5.16) to evaluate $\mathrm{var}\left(\|\tilde{\boldsymbol{w}}_k\|^2\right)$. Thus, in principle, we know how to evaluate the ratio $\rho(k)$ in the vector case. A drawback of this method is that the matrix Ψ lies in $\mathbb{R}^{M^4\times M^4}$, and it becomes difficult to solve the recursion of Theorem 5.6 explicitly for filter lengths larger than $M = 6$. If the entries of $\boldsymbol{x}_k$ are mutually independent, several elements of Ψ vanish, and sparse matrix techniques can be used to simplify the problem.

In any case, our recursions allow us to evaluate $\rho(k)$ (as defined in (5.15)). An example with Gaussian inputs and $M = 2$ is shown in Fig, 5.10 with the curves for both $\rho(k)$ and $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2$ with $R = I$ and $\mu = 0.25$. The value of μ is chosen to be close to the value that achieves fastest convergence of $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2$ in this case. Notice that, as in the scalar case, the simulation shows $\rho(k)$ growing with k. It also shows that $\rho(k)$ assumes relatively small values at the beginning of the simulation so that there will be good agreement between the actual learning curve and the ensemble-average learning curve for small k. Figure 5.12 further ahead confirms this effect for filters of length $M = 100$.

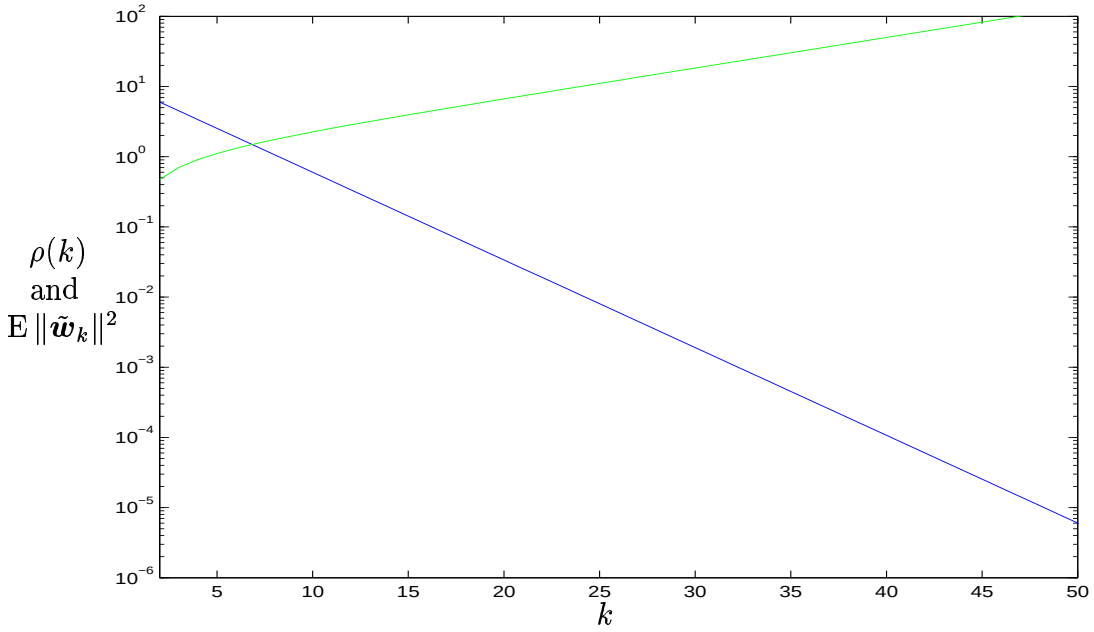

Figure 5.10: $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^2$ *(dark curve) and* $\rho(k)$ *(light curve), computed for* $M = 2$, $\mu = 0.25$, *and* $R = I$.

5.4.2 Almost-Sure Convergence: Solution for a Simplified Model

As we mentioned in the scalar case, the variance analysis explains reasonably well the initial behavior of the ensemble-average learning curve but it cannot predict the behavior for large k. For that we need an almost-sure convergence analysis similar to what we did in Sec. 5.3.2. We start by considering a simplified model here that will show that the effects we observed before still exist in the vector case. It will also show that some new effects arise, especially the sensitivity of the behavior of the ensemble-average learning curve to the direction of the initial condition. In Sec. 5.4.3, we shall present a method of analysis that applies to more general models and input distributions.

So let $\boldsymbol{e}^{(i)}$ represent the i-th basis vector, i.e., $e_j^{(i)} = 1$ if $i = j$ and zero otherwise, and assume that the input sequence $\{\boldsymbol{x}_k\}$ is of the form

$$\boldsymbol{x}_k = r(k)\boldsymbol{s}_k, \tag{5.18}$$

where $r(k)$ is a random variable with zero mean. The vector $\boldsymbol{s}_k$ is independent of $r(k)$ and satisfies

$$\boldsymbol{s}_k = \boldsymbol{e}^{(i)}, \quad \text{with probability } p_i.$$

In other words, $\boldsymbol{x}_k$ may assume only one out of M orthogonal directions. Note that the entries of $\boldsymbol{x}_k$ are *dependent* in this case. As we did before, we assume that the noise is identically zero.

With these definitions, the weight vector $\tilde{\boldsymbol{w}}_{k+1}$ is given by

$$\tilde{\boldsymbol{w}}_{k+1} = \prod_{i=0}^{k} \left(I - \mu \boldsymbol{x}_i \boldsymbol{x}_i^T\right) \tilde{\boldsymbol{w}}_0 =$$

$$= \begin{bmatrix} \prod_{i=0}^{k} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(1)}\right] & & \\ & \ddots & \\ & & \prod_{i=0}^{k} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(M)}\right] \end{bmatrix} \tilde{\boldsymbol{w}}_0.$$

Using this relation, we can compute $\|\tilde{\boldsymbol{w}}_k\|^2$ and $e(k)^2$ as follows.

$$\|\tilde{\boldsymbol{w}}_k\|^2 = \sum_{l=1}^{M} \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \tilde{w}_{0,\,l}^2, \tag{5.19}$$

$$e(k)^2 = \begin{cases} r(k)^2 \tilde{w}_{0,\,1}^2 \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(1)}\right]^2, & \text{with probability } p_1, \\ \quad \vdots & \quad \vdots \\ r(k)^2 \tilde{w}_{0,\,M}^2 \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(M)}\right]^2, & \text{with probability } p_M. \end{cases} \tag{5.20}$$

Mean-square analysis

Let $\mathrm{E}\, r(i)^2 = \sigma_2$, and $\mathrm{E}\, r(i)^4 = \sigma_4$. Since all $\boldsymbol{s}_i$ and $r(i)$ are independent and $\mathrm{E}\, \boldsymbol{s}_i^T \boldsymbol{e}^{(l)} = p_l$, the MSD and MSE are given by

$$\mathrm{E}\, \|\tilde{\boldsymbol{w}}_k\|^2 = \sum_{l=1}^{M} \left(1 - 2\mu\sigma_2 p_l + \mu\sigma_4 p_l\right)^k \tilde{w}_{0,\,l}^2, \tag{5.21}$$

$$\mathrm{E}\, e(k)^2 = \sigma_2 \sum_{l=1}^{M} p_l \left(1 - 2\mu\sigma_2 p_l + \mu\sigma_4 p_l\right)^k \tilde{w}_{0,\,l}^2. \tag{5.22}$$

These relations express the MSD and the MSE in terms of exponential terms that depend on the factors $(1 - 2\mu\sigma_2 p_l + \mu\sigma_4 p_l)$, which are equal to

$$(1 - 2\mu\sigma_2 p_l + \mu\sigma_4 p_l) \;=\; \mathrm{E}\left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \;.$$

Therefore, the MS convergence of all the modes will require that μ be such that

$$\ln \mathrm{E}\left[1-\mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \;<\; 0\,, \tag{5.23}$$

or, equivalently, $\ln \mathrm{E}\,[1-2p_l\mu\sigma_2+p_l\mu^2\sigma_4\,]<0$ for all $1\leq l\leq M$.

Almost-sure analysis

Consider now one of the products in (5.19), *i.e.,*

$$P_l \stackrel{\Delta}{=} \prod_{i=0}^{k-1}\left[1-\mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \tilde{w}_{0,\,l}^2\,.$$

Since this product has the same form as (5.2) in the scalar case, we can use our results of Sec. 5.3.2 to analyze its behavior. Evaluating the logarithm of P_l we obtain

$$\ln P_l = \sum_{i=0}^{k-1} \ln\left[1-\mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 + \ln \tilde{w}_{0,\,l}^2\,,$$

and, as before,

$$\frac{\ln P_l}{k} \xrightarrow{a.s.} \mathrm{E}\ln\left[1-\mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 = p_l\,\mathrm{E}\ln\left[1-\mu\, r(i)^2\right]^2.$$

Following the discussion in Sec. 5.3.2, we conclude that almost-sure convergence requires (in contrast to (5.23)),

$$\mathrm{E}\ln\left[1-\mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \;<\; 0\,, \tag{5.24}$$

or, equivalently, $\mathrm{E}\ln\big(1-\mu\, r(i)^2\big)^2<0$, so that each term of (5.19) will eventually (for large k) converge to zero at a rate that is determined by

$$\lambda_l = e^{p_l\,\mathrm{E}\ln\left(1-\mu\, r(i)^2\right)^2},$$

as opposed to the MS rate of $\big(1-2\mu\sigma_2 p_l+\mu^2\sigma_4 p_l\big)$ given by (5.21). The distinction between conditions (5.23) and (5.24) highlights again the same phenomenon that

occurred in the scalar case, *viz.,* for large k, the rates of convergence of the true learning curve and the ensemble-average learning curve will be distinct, with the latter decaying faster.

Sensitivity to the initial condition

A new feature of the vector case is that the behavior of $\|\tilde{\boldsymbol{w}}_k\|^2$ is now dependent on the direction of the initial condition $\tilde{\boldsymbol{w}}_0$. Indeed, assume for example that all the probabilities p_l are equal, *i.e.*, $p_l = 1/M$, and that all the entries of the vector $\tilde{\boldsymbol{w}}_0$ are also equal. To further simplify the discussion, normalize $\tilde{\boldsymbol{w}}_0$ so that $\|\tilde{\boldsymbol{w}}_0\| = 1$, that is, we choose $\tilde{\boldsymbol{w}}_{0,l} = \pm 1/\sqrt{M}$. In this situation, the norm of the weight error vector becomes

$$\|\tilde{\boldsymbol{w}}_k\|^2 = \frac{1}{M} \sum_{l=1}^{M} \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2, \tag{5.25}$$

where the distribution of each of the terms in the sum is exactly the same. This means that $\|\tilde{\boldsymbol{w}}_k\|^2$ is in fact an average of M (*not* independent) scalar learning curves, each described by a term of the form

$$\tilde{q}_{k,l} \;=\; \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(l)}\right]^2 \tilde{q}_{0,l}\;, \quad \tilde{q}_{0,l} = 1\;.$$

That is,

$$\|\tilde{\boldsymbol{w}}_k\|^2 = \frac{1}{M} \sum_{l=1}^{M} \tilde{q}_{k,l}\;.$$

Therefore, we should expect the variance of $\|\tilde{\boldsymbol{w}}_k\|^2$ to be smaller than that of each term in the sum (as we saw in (5.12) and in Sec. 5.3.5).

On the other hand, if (for example) $\tilde{\boldsymbol{w}}_{0,1} = 1$ and $\tilde{\boldsymbol{w}}_{0,2} = \cdots = \tilde{\boldsymbol{w}}_{0,M} = 0$, then

$$\|\tilde{\boldsymbol{w}}_k\|^2 = \prod_{i=0}^{k-1} \left[1 - \mu\, r(i)^2 \boldsymbol{s}_i^T \boldsymbol{e}^{(1)}\right]^2. \tag{5.26}$$

Since there is no averaging effect in the computation of the norm anymore, we should see exactly the same kind of behavior as for the scalar LMS algorithm. For other values of the initial condition $\tilde{\boldsymbol{w}}_0$, we have an averaging effect between the extremes of (5.25) and (5.26).

In Fig. 5.11 we plot the curves $\mathrm{E}\ln(1-\mu\, r^2)^2$ and $\ln \mathrm{E}(1-\mu\, r^2)^2$ for a variable r^2 that is distributed as a $\mathcal{X}^2$ variable with 100 degrees of freedom. This is exactly the distribution of $\|\boldsymbol{y}\|^2$ if the entries of the random vector $\boldsymbol{y} \in \mathbb{R}^{100}$ are Gaussian independent variables with unit variance [Cra46, p. 233]. The probability density function of a $\mathcal{X}^2$ random variable z with M degrees of freedom is

$$p(z) = \frac{1}{2^{M/2}\Gamma(M/2)}(z)^{\frac{M}{2}-1}e^{-z/2}, \tag{5.27}$$

where $\Gamma(\cdot)$ is the Gamma function. For a variable r with this distribution, we have $\sigma_2 = \mathrm{E}\, r^2 = M$ and $\sigma_4 = \mathrm{E}\, r^4 = M^2 + 2M$. Assuming that $p_1 = p_2 = \cdots = p_M = 1/M$, we conclude from (5.23) that

$$\ln \mathrm{E}(1-\mu r^2)^2 = \ln\Big(1 - 2\mu + \big(M+2\big)\mu^2\Big).$$

Summarizing, the above discussion shows that an LMS filter with length $M > 1$ and with input satisfying (5.18) will behave in a manner similar to that of a scalar LMS filter for which $\boldsymbol{x}_k$ has the same probability distribution as $r(n)$, but with two main differences:

1. The rate of convergence will now be smaller (depending on the values of the probabilities p_l).

2. A single realization of the error $\|\tilde{\boldsymbol{w}}_k\|^2$ will tend to be close to its mean $\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2$ for a longer time, because of the averaging performed when computing the norm (5.25).

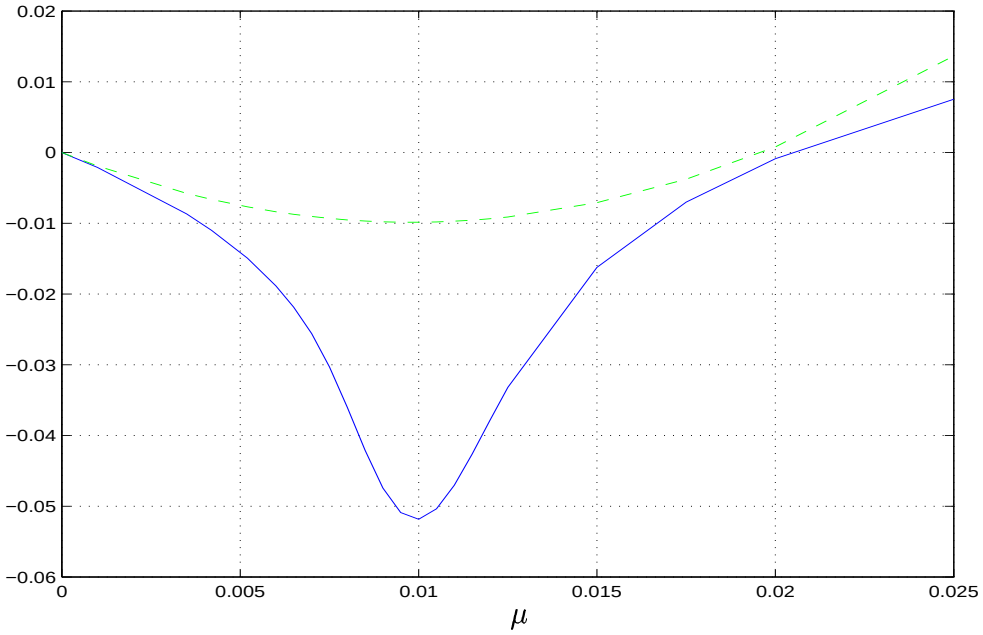

Figure 5.11: *Graphs of* $\mathrm{E}\ln\left(1-\mu\, r^2\right)^2$ *(continuous line) and* $\ln \mathrm{E}\left(1-\mu\, r^2\right)^2$ *(broken line), for* $\mathcal{X}^2$ *distribution with* 100 *degrees of freedom.*

Fig. 5.12 illustrates the above results for a filter with 100 taps. In the first figure, all entries of the initial condition $\tilde{\boldsymbol{w}}_0$ are equal, while only the first entry $\tilde{w}_{0,1}$ in the initial condition for Fig. 5.12(b) was nonzero. Both figures show ensemble-average learning curves computed with $L = 1,000$. Note how the first simulation stays close to $\mathrm{E}\, e(k)^2$ for a longer time, as we predicted above. Note, though, that in both simulations the ensemble-average learning curves eventually tend to decrease with the (fastest) rate predicted by almost-sure analysis (which, in this case, is equal to 0.9646, while the (slowest) rate predicted by MS analysis is 0.9905). In the simulations, $r^2(k)$ was distributed as a $\mathcal{X}^2$ variable with 100 degrees of freedom.

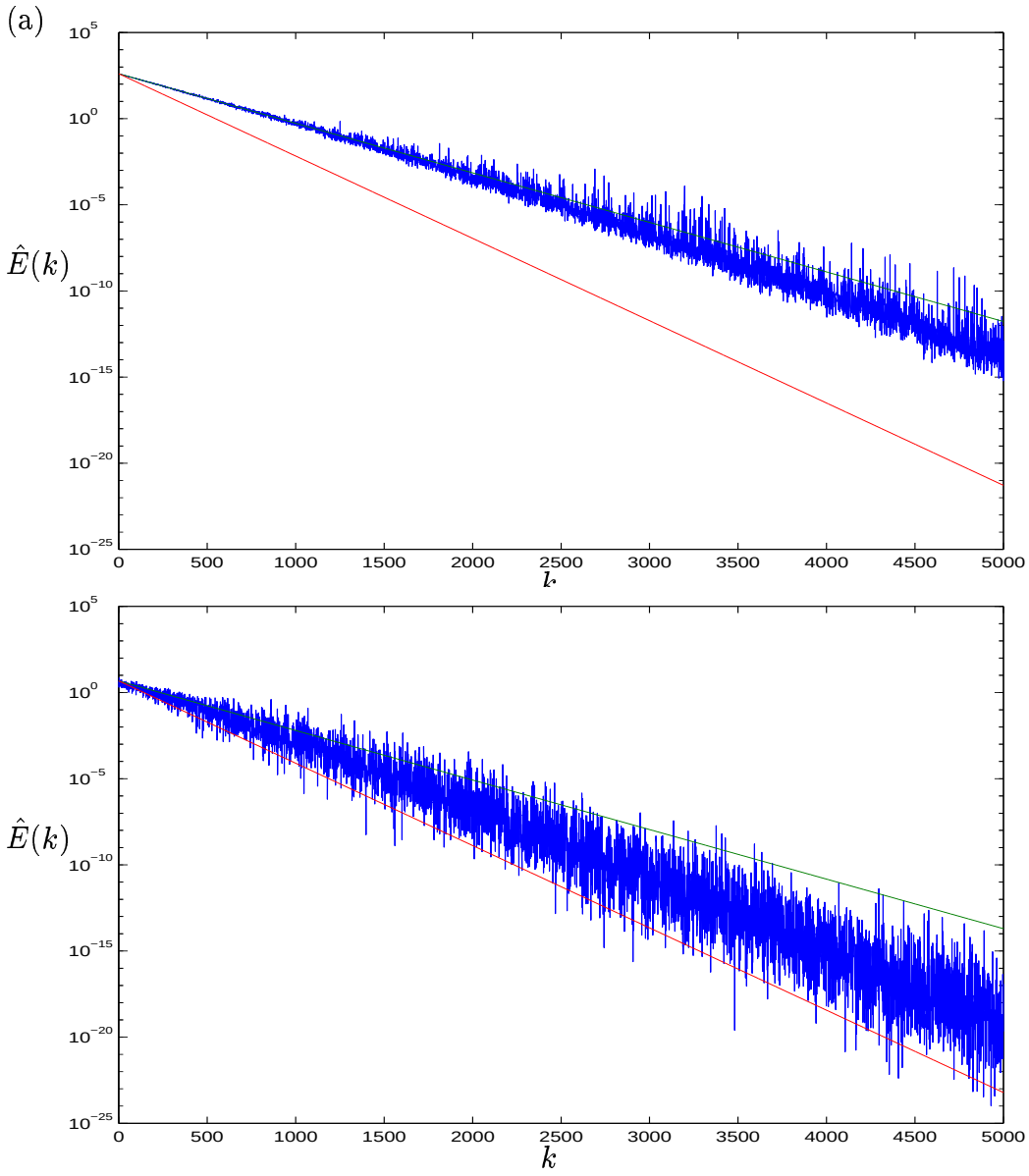

Figure 5.12: $\hat{E}(k)^2$ *computed with* $L = 1000$, $M = 100$ *and* $\mu = 0.0042$. *The input sequence satisfies* (5.18), *and* r^2 *is a* $\mathcal{X}^2$ *random variable with mean* 100 *and* 100 *degrees of freedom. (a) All entries of the initial condition* $\tilde{\boldsymbol{w}}_0$ *are equal. (b) Only the first entry of the initial condition* $\tilde{\boldsymbol{w}}_0$ *is nonzero. The upper smooth curve is* $\mathrm{E}\, e(k)^2$ *computed theoretically, and the lower curve is the rate of convergence predicted by a.s. convergence analysis.*

5.4.3 Almost-Sure Convergence: A Solution for General Models

The analysis in the previous section assumed a special regression sequence $\{\boldsymbol{x}_k\}$. Although restrictive, the resulting simplified model showed that the effects we observed in the scalar case still occur in the vector case. We now provide an analysis that applies to general regression vectors $\boldsymbol{x}_k$ and which allows us to further highlight the distinction that exists between the true learning curve and an ensemble-average learning curve.

Thus using the LMS error equation we obtain

$$\tilde{\boldsymbol{w}}_k = \left[\prod_{i=0}^{k-1}(I - \mu \boldsymbol{x}_i \boldsymbol{x}_i^T)\right] \tilde{\boldsymbol{w}}_0 \triangleq \Phi_k \tilde{\boldsymbol{w}}_0 \ ,$$

where we defined the state-transition matrix Φ_k. In the simplified model prior to (5.19), the matrix Φ_k was assumed diagonal, which led to (5.19). Now we get

$$\|\tilde{\boldsymbol{w}}_k\|^2 = \tilde{\boldsymbol{w}}_0^T \Phi_k^T \Phi_k \tilde{\boldsymbol{w}}_0.$$

The rate of convergence of $\|\tilde{\boldsymbol{w}}_k\|^2$ will therefore be dependent on the eigenvalues of $\Phi_k^T \Phi_k$. For the simple model (5.18) of the previous section, we were able to determine the properties of each individual eigenvalue of $\Phi_k^T \Phi_k$. In order to extend the analysis to more general input distributions, we study in this section the evolution of the determinant of $(\Phi_k^T \Phi_k)$, *i.e.*, we now study the product of the eigenvalues of $\Phi_k^T \Phi_k$, and compare this product with $\det \mathrm{E}\big(\Phi_k^T \Phi_k\big)$, since

$$\mathrm{E}\,\|\tilde{\boldsymbol{w}}_k\|^2 = \tilde{\boldsymbol{w}}_0^T\, \mathrm{E}\big(\Phi_k^T \Phi_k\big) \tilde{\boldsymbol{w}}_0.$$

Mean-square analysis

The computation of $\det\big(\mathrm{E}\,\Phi_k^T \Phi_k\big)$ can be performed in the case of iid input regressors $\{\boldsymbol{x}_k\}$ by using our recursion for $\mathrm{vec}(\bar{C}_k)$ (2.28). Indeed,

$$\det\big(\mathrm{E}\,\Phi_k^T \Phi_k\big) = \det \mathrm{E}\Big[\big(I - \mu \boldsymbol{x}_0 \boldsymbol{x}_0^T\big) \ldots \big(I - \mu \boldsymbol{x}_{k-1} \boldsymbol{x}_{k-1}^T\big)\big(I - \mu \boldsymbol{x}_{k-1} \boldsymbol{x}_{k-1}^T\big) \ldots \big(I - \mu \boldsymbol{x}_0 \boldsymbol{x}_0^T\big)\Big].$$

On the other hand, from (2.28) we obtain

$$\bar{C}_k = \mathrm{E}\,\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T = \mathrm{E}\left[\left(I-\mu\boldsymbol{x}_{k-1}\boldsymbol{x}_{k-1}^T\right)\cdots\left(I-\mu\boldsymbol{x}_0\boldsymbol{x}_0^T\right)\bar{C}_0\left(I-\mu\boldsymbol{x}_0\boldsymbol{x}_0^T\right)\cdots\left(I-\mu\boldsymbol{x}_{k-1}\boldsymbol{x}_{k-1}^T\right)\right].$$

This recursion can be solved using Kronecker products, as we showed in (2.28). We can use this same result to compute $\det\left(\mathrm{E}\,\Phi_k\Phi_k^T\right)$ as follows. Let F_k be obtained from the above recursion, but with $\bar{C}_0$ replaced by the identity matrix. Since $\{\boldsymbol{x}_k\}$ is stationary and iid, the order of the matrices in the product is irrelevant, and $\mathrm{E}\,\Phi_k\Phi_k^T = \mathrm{E}\,\Phi_k^T\Phi_k$. Therefore, we have

$$\det\left(\mathrm{E}\,\Phi_k^T\Phi_k\right) = \det F_k, \tag{5.28}$$

where $\mathrm{vec}(F_k)$ satisfies

$$\mathrm{vec}(F_k) = \left[I_{M^2} - \mu\left(R\otimes I_M\right) - \mu\left(I_M\otimes R\right) + \mu^2\,\mathrm{E}\left(\boldsymbol{x}_{k-1}\boldsymbol{x}_{k-1}^T\otimes\boldsymbol{x}_{k-1}\boldsymbol{x}_{k-1}^T\right)\right]\mathrm{vec}(F_{k-1}),$$

with initial condition $F_0 = I$.

We will shortly present an example where the computation of $\det\left(\mathrm{E}\,\Phi_k^T\Phi_k\right)$ simplifies, and a simple formula for its rate of convergence can be obtained.

Almost-sure determinant analysis

The determinant of Φ_k satisfies

$$\det\Phi_k = \prod_{i=1}^{k-1}\left[\det\left(I-\mu\boldsymbol{x}_i\boldsymbol{x}_i^T\right)\right] = \prod_{i=0}^{k-1}\left(1-\mu\|\boldsymbol{x}_i\|^2\right),$$

where we used the fact that the matrix $I-\mu\boldsymbol{x}_i\boldsymbol{x}_i^T$ has $M-1$ eigenvalues at 1 and one eigenvalue at $1-\mu\|\boldsymbol{x}_i\|^2$. We then obtain

$$\det\left(\Phi_k^T\Phi_k\right) = \prod_{i=0}^{k-1}\left(1-\mu\|\boldsymbol{x}_i\|^2\right)^2,$$

which has the same form as a scalar LMS algorithm with input sequence $\{\|\boldsymbol{x}_k\|\}$, so that

$$\frac{1}{k}\ln\det\left(\Phi_k\Phi_k^T\right) \xrightarrow{a.s.} \mathrm{E}\ln\left(1-\mu\|\boldsymbol{x}_i\|^2\right)^2 .$$

Therefore, all of our previous results can be directly applied to this case. In particular, the rate of convergence (or divergence) of $\det \Phi_k^T \Phi_k$ for large k is almost surely given by

$$e^{\mathrm{E} \ln \left(1-\mu \|\boldsymbol{x}_i\|^2\right)^2},$$

which in general will be different than the rate obtained from (5.28).

To explicitly find the a.s. rate of convergence, it is necessary to know the distribution of $\|\boldsymbol{x}_k\|^2$, which depends on the distribution of $\boldsymbol{x}_k$ itself. We consider a few special cases below.

For example, let $\{\boldsymbol{x}_k\}$ be iid and such that the entries of each vector are mutually independent and Gaussian with unit variance. We saw in the previous section that in this case, $\|\boldsymbol{x}_k\|^2$ is distributed as a $\mathcal{X}^2$ with M degrees of freedom (5.27). In this case, the computation of $\det \mathrm{E}\, \Phi_k^T \Phi_k$ simplifies considerably, as follows.

$$\mathrm{E}\left(I - \mu \boldsymbol{x}_i \boldsymbol{x}_i^T\right)^2 = \Big(1 - 2\mu + (M+2)\mu^2\Big) I,$$

Since this is a multiple of the identity, $\det \mathrm{E}\left(\Phi_k^T \Phi_k\right)$ reduces to

$$\det\left(\mathrm{E}\, \Phi_k^T \Phi_k\right) = \left(1 - 2\mu + (M+2)\mu^2\right)^{Mk}.$$

This is similar to the expression that we obtained for the simplified model of Sec. 5.4.2, Fig. 5.11, except that the factor $(1 - 2\mu + (M+2)\mu^2)$ is now raised to the power M. This means that for $M = 100$, the plots of Fig. 5.11 (with the vertical scale multiplied by 100) also apply to this example. Note that this example and that of Sec. 5.4.2 are in fact very different — in this section, $\boldsymbol{x}_k$ may take *any* direction in $\mathbb{R}^M$, unlike what happened in the previous example. It only happens that the determinants have the same properties in both situations.

As another example for the computation of $\mathrm{E}(1-\mu\|\boldsymbol{x}_l\|^2)^2$, assume that the entries of $\boldsymbol{x}_k$ have the same (non-Gaussian) distribution and are independent. In this situation, we can use the Central Limit Theorem [Dur96, p. 112] to conclude that for large M the distribution of $\|\boldsymbol{x}_k\|^2$ will be approximately Gaussian, with mean $M\sigma_2$ and variance $M\left(\sigma_{x,4}-\sigma_{x,2}^2\right)+M^2\sigma_{x,2}^2$, where $\sigma_{x,2}$ and $\sigma_{x,4}$ are respectively the variance and the fourth moment of each entry of $\boldsymbol{x}_k$. This is true as long as both $\sigma_{x,2}$ and $\sigma_{x,4}$ are finite.

5.5 CONTRIBUTIONS OF THIS CHAPTER

We have shown that there are situations in which the behavior of the LMS errors is significantly different than that of their averages. These situations arise when one uses larger step-sizes (i.e., non-infinitesimal) to obtain faster convergence. Our analysis and our simulations show that in some cases, it may be necessary to average a significantly large number of simulations to obtain a good approximation to the mean-square behavior of an adaptive filter. In particular, one must be careful when analyzing ensemble-average learning curves.

Looking at the same results from another perspective, we might conclude that, for larger step-sizes, the average performance *alone* is not sufficient for design purposes.

We have proven our claims analytically, and studied the behavior of the scalar LMS algorithm in detail. For the vector case, we provided a detailed analysis of the variance of $\|\tilde{\boldsymbol{w}}_k\|^2$, and an almost-sure analysis for a particular model of the input sequence $\boldsymbol{x}_k$ and for the determinant of the state-transition matrix Φ_k.

Although our analysis was performed only for the LMS algorithm, one should expect to observe similar behavior in several other gradient-based algorithms,

such as the normalized LMS [Ber86], signed LMS [Ewe94], leaky LMS [MA97], CMA, [God80], etc.

There are related works in the literature that have also studied the almost-sure stability of LMS (e.g., [BA81, BAN86, Sol97]), or even of continuous-time systems (e.g., [Koz69, PE78]). These works, however, tend to assume that almost-sure stability implies reasonable algorithm performance (see, in particular, the recent reference [Sol97]). We have shown in this chapter that this is not the case. An almost-sure stable filter might have very poor performance when it is not also mean-square stable — since we have shown that, for small time, the ensemble-average learning curves tend to stay reasonably close to the (mean-square) theoretical learning curve, an almost-sure stable, but mean-square unstable algorithm would likely have its error diverging to a large value before starting to converge (see Fig. 5.6 and Sec. 5.3.6).

None of the cited works highlights the relation between almost-sure and mean-square rates of convergence with actual, observed ensemble-average learning curves (the one reference that compares mean-square stability and almost-sure stability, [BAN86], does so only to show that they agree for $\mu \approx 0$). Prior to the papers on which this chapter is based, [NS98b, NS99a], there were no results on how, more generally, the notions of mean-square and almost-sure stability compare for non-infinitesimal values of μ, and on how their differences provide a natural explanation for the fact that LMS has two distinct rates of convergence. In particular, our variance analysis in Secs. 5.3.6 and 5.4.1 to explain the appearance of the two rates of convergence is original.

APPENDICES FOR CHAPTER 5

5.A PROOF OF THEOREM 5.3

In this appendix we prove the statement of Thm. 5.3. We do so by showing that both $\mathrm{E}\ln\bigl(1-\mu\boldsymbol{x}_n^2\bigr)^2$ and $\ln\mathrm{E}\bigl(1-\mu\boldsymbol{x}_n^2\bigr)^2$ are differentiable with respect to μ at $\mu=0$, and that both derivatives are equal at that point.

The derivative of the second function evaluates to

$$\frac{\mathrm{d}}{\mathrm{d}\,\mu}\ln\bigl(1-\mu\sigma_2+\mu^2\sigma_4\bigr)\Big|_{\mu=0} = -\left.\frac{\sigma_2}{1-\mu\sigma_2+\mu^2\sigma_4}\right|_{\mu=0} = -2\sigma_2\ .$$

The evaluation of the other derivative is more involved, and will be obtained in several steps in the lemmas below. The first lemma proves that $\mathrm{E}\ln\bigl(1-\mu x^2\bigr)^2$ is well defined for all $\mu\geq 0$.

Lemma 5.A.1. *Under the conditions of Thm. 5.3, the expected value*

$$\mathrm{E}\ln\bigl(1-\mu x^2\bigr)^2 = \int_{-\infty}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x$$

exists and is finite for all $\mu\geq 0$.

Proof: Let δ be a positive constant such that $1/\sqrt{\mu}+\delta > B$, and split the above integral as below

$$\begin{aligned}\mathrm{E}\ln\bigl(1-\mu x^2\bigr)^2 = &\int_{-\infty}^{-\frac{1}{\sqrt{\mu}}-\delta}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x + \int_{-\frac{1}{\sqrt{\mu}}-\delta}^{\frac{1}{\sqrt{\mu}}+\delta}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x + \\ &+\int_{\frac{1}{\sqrt{\mu}}+\delta}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x \triangleq I_1+I_2+I_3.\end{aligned}$$

We can find an upper bound for the middle term using the assumption that $\sup_x p(x)<\infty$, as follows

$$I_2 \leq \int_{-\frac{1}{\sqrt{\mu}}-\delta}^{\frac{1}{\sqrt{\mu}}+\delta}\left|\ln\bigl(1-\mu x^2\bigr)^2 p(x)\right|\mathrm{d}\,x \quad\leq\quad 2\sup_x p(x)\int_0^{\frac{1}{\sqrt{\mu}}+\delta}\left|\ln\bigl(1-\mu x^2\bigr)^2\right|\mathrm{d}\,x.$$

Note that $\ln\left(1-\mu x^2\right)^2$ is non-positive if $0 \le x \le \sqrt{\frac{2}{\mu}}$, and positive otherwise. Therefore, the absolute value in the above integral reduces to

$$2 \sup_x p(x) \int_0^{\frac{1}{\sqrt{\mu}}+\delta} \left|\ln\left(1-\mu x^2\right)^2\right| \mathrm{d}\, x = -2 \sup_x p(x) \int_0^{\sqrt{\frac{2}{\mu}}} \ln\left(1-\mu x^2\right)^2 \mathrm{d}\, x + \\ +2 \sup_x p(x) \int_{\sqrt{\frac{2}{\mu}}}^{\infty} \ln\left(1-\mu x^2\right)^2 \mathrm{d}\, x.$$

Since

$$\int \ln\left(1-\mu x^2\right)^2 \mathrm{d}\, x = \left(x - \frac{1}{\sqrt{\mu}}\right) \ln\left(1-\sqrt{\mu}x\right)^2 + \\ + \left(x + \frac{1}{\sqrt{\mu}}\right) \ln\left(1+\sqrt{\mu}x\right)^2 - 4x,$$

and

$$\lim_{x \to \frac{1}{\sqrt{\mu}}} \left(x - \frac{1}{\sqrt{\mu}}\right) \ln\left(1-\sqrt{\mu}x\right)^2 = 0,$$

it follows that I_2 is finite.

Consider I_3 now. Using the fact that $1/\sqrt{\mu} + \delta > B$ and the lemma's hypotheses, we have

$$I_3 \le \int_{\frac{1}{\sqrt{\mu}}+\delta}^{\infty} \left|\ln\left(1-\mu x^2\right)^2 p(x)\right| \mathrm{d}\, x < \int_{\frac{1}{\sqrt{\mu}}+\delta}^{\infty} \left|\frac{\ln\left(1-\mu x^2\right)^2}{x^\beta}\right| \mathrm{d}\, x.$$

Since $\ln\left(1-\mu x^2\right)^2 < x$ for all positive x and $\beta > 5$, the above integral is finite. The last term, I_1, can be bounded in a similar manner.

$\diamondsuit$

With a small modification, the same arguments can be used to prove that $\mathrm{var}\left(\mathrm{E}\ln\left(1-\mu x^2\right)^2\right)$ is finite.

Having proved that $\mathrm{E}\ln\bigl(1-\mu x^2\bigr)^2$ exists, we now show that this function is differentiable at $\mu = 0$. Unfortunately, we cannot simply apply the formula

$$\frac{\mathrm{d}}{\mathrm{d}\,\mu}\int_{-\infty}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x = \int_{-\infty}^{\infty}\frac{\partial\ln\bigl(1-\mu x^2\bigr)^2}{\partial\mu}p(x)\,\mathrm{d}\,x,$$

because $\ln\bigl(1-\mu\boldsymbol{x}_n^2\bigr)^2$ is not a bounded function and its derivative is not integrable, except at $\mu = 0$ [Rud76, pp. 236–239]. We need to compute the derivative of $\mathrm{E}\ln\bigl(1-\mu\boldsymbol{x}_n^2\bigr)^2$ directly from the definition, that is, we shall show that

$$\lim_{\mu\to 0}\frac{\int_{-\infty}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x}{\mu} - \int_{-\infty}^{\infty}\bigl(-2x^2\bigr)p(x)\,\mathrm{d}\,x = 0.$$

The computation of the above limit is carried out in the three lemmas below. The first two results show that we can avoid the singular points at $x = \pm\frac{1}{\sqrt{\mu}}$ by restricting the integration limits to $-\mu^{-1/8}$ and $\mu^{-1/8}$.

Lemma 5.A.2. *Assume that the conditions of Thm. 5.3 hold, and that μ satisfies*

$$\mu < \min\left\{\frac{1}{10},\,\frac{1}{2B^2}\right\}. \tag{5.A.1}$$

Then there exists a finite constant C_1 such that

$$\left|\int_{-\infty}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x - \int_{-\mu^{-1/8}}^{\mu^{-1/8}}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x\right| < C_1\mu^{\frac{\beta}{8}}. \tag{5.A.2}$$

Proof: Using (5.A.1) we can upper bound the above difference as

$$\left|\int_{-\infty}^{-\mu^{-1/8}}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x + \int_{\mu^{-1/8}}^{\infty}\ln\bigl(1-\mu x^2\bigr)^2 p(x)\,\mathrm{d}\,x\right| <$$

$$< 2\int_{\mu^{-1/8}}^{\infty}\left|\frac{\ln\bigl(1-\mu x^2\bigr)^2}{x^\beta}\right|\mathrm{d}\,x.$$

Performing the change of variables $y = \sqrt{\mu}\,x$, we obtain

$$2\int_{\mu^{-1/8}}^{\infty}\left|\frac{\ln\bigl(1-\mu x^2\bigr)^2}{x^\beta}\right|\mathrm{d}\,x = 2\mu^{\frac{\beta-1}{2}}\int_{\mu^{3/8}}^{\infty}\left|\frac{\ln\bigl(1-y^2\bigr)^2}{y^\beta}\right|\mathrm{d}\,y <$$

$$< 2\mu^{\frac{\beta-1}{2}}\left[\int_{\mu^{3/8}}^{0.5}\frac{|\ln 0.75|}{y^\beta}\,\mathrm{d}\,y + \int_{0.5}^{\infty}\left|\frac{\ln\bigl(1-y^2\bigr)^2}{y^\beta}\right|\mathrm{d}\,y\right].$$

Using an argument similar to that of the previous lemma, we can show that the last integral is finite, i.e.,

$$\int_{0.5}^{\infty} \left| \frac{\ln\left(1-y^2\right)^2}{y^\beta} \right| \mathrm{d}\, y = C_{11} < \infty.$$

The other integral evaluates to

$$\int_{\mu^{3/8}}^{0.5} \frac{\left|\ln 0.75\right|}{y^\beta} \mathrm{d}\, y = \frac{2\left|\ln 0.75\right|}{\beta-1}\left(\mu^{-\frac{3}{8}(\beta-1)} - 2^{\beta-1}\right) \triangleq C_{12}\mu^{-\frac{3}{8}(\beta-1)} + C_{13}.$$

Inequality (5.A.2) follows from these two results, and from the fact that $\mu^{\frac{\beta-1}{2}} < \mu^{\frac{\beta}{8}}$ for $\mu < 1$.

♢

Lemma 5.A.3. *The inequality below is satisfied under the conditions of the previous lemma,*

$$\left| -2\int_{-\infty}^{\infty} x^2 p(x)\, \mathrm{d}\, x + \int_{-\mu^{-1/8}}^{\mu^{-1/8}} 2x^2 p(x)\, \mathrm{d}\, x \right| < C_2 \mu^{\frac{\beta-3}{8}},$$

where C_2 is a finite constant.

Proof: From (5.A.1), and since $\beta > 5$, we have

$$2\left| -\int_{-\infty}^{\infty} x^2 p(x)\, \mathrm{d}\, x + \int_{-\mu^{-1/8}}^{\mu^{-1/8}} x^2 p(x)\, \mathrm{d}\, x \right| < 4\left| \int_{\mu^{-1/8}}^{\infty} \frac{x^2}{x^\beta}\, \mathrm{d}\, x \right| = \frac{4}{\beta-3}\mu^{\frac{\beta-3}{8}}.$$

♢

Up to now, we have shown that

$$\left| \int_{-\infty}^{\infty} \left[\frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2 \right] p(x)\, \mathrm{d}\, x \right| < \left| \int_{-\frac{1}{2\sqrt{\mu}}}^{\frac{1}{2\sqrt{\mu}}} \left[\frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2 \right] p(x)\, \mathrm{d}\, x \right| + \\ + \left(C_1 + C_2\right)\mu^{\frac{\beta-3}{8}}. \tag{5.A.3}$$

Although $\frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2$ converges to zero as $\mu \to 0$, in order to bound the above integral we need to find how the convergence depends on x. This is done in the next lemma.

Lemma 5.A.4. *The following inequality holds for all x and sufficiently small μ:*

$$0 > \frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2 > -\frac{2x^2}{\frac{1}{\mu x^2} - 1}. \tag{5.A.4}$$

Proof: First note that

$$\frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2 = \ln\left[\frac{\left(1-\mu x^2\right)^{2/\mu}}{e^{-2x^2}}\right]. \tag{5.A.5}$$

We find a bound for this function by studying the convergence of $(1-a/n)^n$ to e^{-a} as $n \to \infty$.

We begin our analysis by noting that the sequence $\left\{\left(1+\frac{1}{n}\right)^n\right\}_{n=0}^{\infty}$ is strictly increasing and upper bounded by $1+1+2^{-1}+2^{-2}+\cdots+2^{-(n-1)}$. This implies that the inequality below holds for $m > n$

$$0 < \left(1+\frac{1}{m}\right)^m - \left(1+\frac{1}{n}\right)^n < \frac{1}{2^n} + \cdots + \frac{1}{2^{m-1}}.$$

Taking this inequality to the limit as $m \to \infty$ and dividing the result by e, we have

$$1 > \frac{\left(1+\frac{1}{n}\right)^n}{e} > 1 - \frac{1}{2^{n-1}e}. \tag{5.A.6}$$

Next, translate this inequality for the case $(1-1/n)^n$, as follows. Consider the limit

$$\lim_{m\to-\infty}\left(1+\frac{1}{m}\right)^m = e,$$

and perform the change of variables $t = -m-1$ to obtain

$$\left(1+\frac{1}{m}\right)^m = \left(1-\frac{1}{t+1}\right)^{-t-1} = \left(1+\frac{1}{t}\right)^t\left(1+\frac{1}{t}\right).$$

Applying (5.A.6) to this relation, we obtain for $m < -1$

$$1 - \frac{1}{m+1} > \frac{\left(1+\frac{1}{m}\right)^m}{e} > \left(1 - 2^{m+2}\right)\left(1 - \frac{1}{m+1}\right) > 1,$$

where the last inequality is true for sufficiently large $|m|$.

Performing the change of variables $m = -n/a$ (for some $a > 0$), we obtain

$$1 + \frac{1}{\frac{n}{a} - 1} > \frac{\left(1-\frac{a}{n}\right)^{\frac{n}{a}}}{e} > 1.$$

Finally, raise these inequalities to the power $-a$ and take the logarithm to obtain

$$0 > \ln\left[\frac{\left(1-\frac{a}{n}\right)^n}{e^{-a}}\right] > -a\ln\left[1 + \frac{1}{\frac{n}{a}-1}\right].$$

We can find an approximation for the logarithm on the right-hand side as follows. Let ε be a small positive number, and expand $\ln(1+\varepsilon)$ in a Taylor series around $\varepsilon = 0$ to find

$$\ln(1+\varepsilon) = \varepsilon - \frac{\varepsilon^2}{2} + \frac{\varepsilon^3}{3} - \cdots < \varepsilon$$

and thus

$$0 > \ln\left[\frac{\left(1-\frac{a}{n}\right)^n}{e^a}\right] > -\frac{a}{\frac{n}{a}-1}.$$

Applying this inequality to (5.A.5) with $a = 2x^2$ and $n = 2/\mu$, we obtain (5.A.4).

$\diamondsuit$

With this result we can bound the remaining integral in (5.A.3) as below. Assume that $\mu^{3/4} < 0.5$, so that

$$\frac{1}{\frac{\mu^{1/4}}{\mu} - 1} < 2\mu^{3/4}.$$

It then follows from Lemma 5.A.4 that

$$\left| \int_{-\frac{1}{\mu^{1/8}}}^{\frac{1}{\mu^{1/8}}} \left(\frac{\ln\left(1-\mu x^2\right)^2}{\mu} + 2x^2 \right) p(x)\,\mathrm{d}\,x \right| < \sup_x p(x) \int_{-\frac{1}{\mu^{1/8}}}^{\frac{1}{\mu^{1/8}}} \frac{2x^2}{\frac{1}{\mu x^2}-1}\,\mathrm{d}\,x <$$

$$< 8 \sup_x p(x) \mu^{3/8}.$$

Substituting this result in (5.A.3), we conclude that

$$\frac{\mathrm{d}}{\mathrm{d}\,\mu} \int_{-\infty}^{\infty} \ln\left(1-\mu x^2\right)^2 p(x)\,\mathrm{d}\,x = -2 \int_{-\infty}^{\infty} x^2 p(x)\,\mathrm{d}\,x = -2\sigma_2,$$

which is our desired result.

5.B AN AUXILIARY RESULT

The following lemma is used in Sec. 5.4.

Lemma 5.B.1. *Let $\boldsymbol{u} \in I\!R^M$ be a column vector. Then*

$$\|\boldsymbol{u}\|^4 = \mathrm{Tr}\left(\boldsymbol{u}\boldsymbol{u}^T\right)^{\otimes 2}.$$

Proof: Note that

$$\|\boldsymbol{u}\|^4 = \left(\boldsymbol{u}^T\boldsymbol{u}\right)^2 = \left(\boldsymbol{u}^T\boldsymbol{u}\right) \mathrm{Tr}\left(\boldsymbol{u}\boldsymbol{u}^T\right) = \left(\sum_{i=1}^{M} u_i^2\right) \mathrm{Tr}\left(\boldsymbol{u}\boldsymbol{u}^T\right).$$

On the other hand, from the definition of Kronecker product we have

$$\left(\boldsymbol{u}\boldsymbol{u}^T\right)^{\otimes 2} = \boldsymbol{u}\boldsymbol{u}^T \otimes \boldsymbol{u}\boldsymbol{u}^T = \begin{bmatrix} u_1^2\boldsymbol{u}\boldsymbol{u}^T & u_1u_2\boldsymbol{u}\boldsymbol{u}^T & \dots & u_1u_M\boldsymbol{u}\boldsymbol{u}^T \\ u_1u_2\boldsymbol{u}\boldsymbol{u}^T & u_2^2\boldsymbol{u}\boldsymbol{u}^T & \dots & u_2u_M\boldsymbol{u}\boldsymbol{u}^T \\ \vdots & \vdots & \ddots & \vdots \\ u_1u_M\boldsymbol{u}\boldsymbol{u}^T & u_2u_M\boldsymbol{u}\boldsymbol{u}^T & \dots & u_M^2\boldsymbol{u}\boldsymbol{u}^T \end{bmatrix},$$

and thus

$$\mathrm{Tr}\left(\boldsymbol{u}\boldsymbol{u}^T \otimes \boldsymbol{u}\boldsymbol{u}^T\right) = \sum_{i=1}^{M}\left(u_i^2\,\mathrm{Tr}\big(\boldsymbol{u}\boldsymbol{u}^T\big)\right) = \|\boldsymbol{u}\|^4.$$

$\diamondsuit$

5.C PROOF OF LEMMA 5.6

Using the LMS recursion with $v(k) \equiv 0$, we can compute $\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T$

$$\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T = \big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big),$$

and therefore

$$\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T \otimes \tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T = \left[\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\right]^{\otimes 2}. \tag{5.C.1}$$

This expression can be simplified using another property of Kronecker products. For any matrices A, B, C, and D, it holds that [HJ94, p. 244]

$$\big(A \otimes B\big)\big(C \otimes D\big) = \big(AC\big) \otimes \big(BD\big). \tag{5.C.2}$$

Apply this property to (5.C.1) with

$$A = B = \big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big), \qquad C = D = \tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big),$$

to obtain

$$\big(\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T\big)^{\otimes 2} = \big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)^{\otimes 2}\left[\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\right]^{\otimes 2}.$$

Applying (5.C.2) again, now with

$$A = B = \tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T, \qquad C = D = \big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big),$$

we obtain

$$\big(\tilde{\boldsymbol{w}}_{k+1}\tilde{\boldsymbol{w}}_{k+1}^T\big)^{\otimes 2} = \big(I-\mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)^{\otimes 2}\big(\tilde{\boldsymbol{w}}_k\tilde{\boldsymbol{w}}_k^T\big)^{\otimes 2}\big(I-\mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)^{\otimes 2}.$$

We can now apply (2.22) and take expected values to obtain the desired result.

CHAPTER 6

THE DRIFT PROBLEM AND FINITE-PRECISION EFFECTS

In most of our prior discussions, we were concerned with how small the steady-state MSE of LMS can be. When finite-precision arithmetic is used, however, a (theoretically) mean-square stable algorithm can have poor performance and even diverge. A useful method of analysis that frees us from concerns about input correlation, and which allows us to study the boundedness or not of the weight estimates, regardless of independence conditions, is the deterministic method of analysis.

In this framework, one essentially studies the stability of an adaptive filter in a worst-case scenario. The results tend of course to be conservative, and the analysis only allows for bounded input signals. Still, the method is capable of handling difficult scenarios, such as finite-precision implementations. In this chapter we shall use it to resolve the aforementioned drift problem of LMS and to propose a new algorithm, called *circular leaky*, which avoids the drawbacks of the classical leakage-based solution (as mentioned in Sec. 2.3).

In particular, we shall show that this new algorithm provides uniformly bounded weight estimates, even if the input-sequence is not persistently exciting (as does the leaky LMS algorithm), but at a reduced computational cost, as compared to leaky LMS. Also, unlike leaky LMS, the estimates computed by the new

algorithm are not biased (this last result is proved using the averaging theorems of Chapter 3). Table 6.1 summarizes the properties of the three different algorithms mentioned above, and also the properties of a modification of the so-called *switching-σ* algorithm used in adaptive control [IS96]. In the complexity column, we list approximate values for the number of multiplications (M), additions (A), multiply-and-accumulate (MA), and if-then (IF) commands necessary for each algorithm.

Algorithm	Drift Problem	Biased when $R > 0$	Complexity			
			MA	M	A	IF
LMS	**YES**	NO	$2M$	1	0	0
Leaky LMS	NO	**YES**	$2M$	$\boldsymbol{M+1}$	0	0
Switching-σ	NO	NO	$\boldsymbol{3M}$	$\boldsymbol{M+2}$	**2**	**3**
Circular Leaky	**NO**	**NO**	$\boldsymbol{2M}$	**3**	**2**	**3**

Table 6.1: *Comparison of the various adaptive algorithms.*

In summary, the original contributions in this chapter include the following:

- A deterministic analysis of a floating-point implementation of LMS.
- The derivation of a new algorithm, circular-leaky, that prevents drift without the introduction of bias, and with a computational cost smaller than that of leaky LMS.
- A stochastic analysis of a modification of the switching-σ algorithm (used in the adaptive control literature).
- Deterministic and stochastic analyses of both circular leaky and switching-σ, when implemented in finite-precision arithmetic.

6.1 THE DRIFT PROBLEM

The fact that the LMS algorithm can produce unbounded weight estimates in some situations is described in several works including, for example, [SLJ86, GMW82, Hay96, IK84]. Reference [SLJ86] studies the drift problem in a deterministic infinite-precision setting, while references [Cio87, CW85] consider finite precision effects. Reference [IK84] provides an analysis in the adaptive control context.

We shall provide here a few examples of drift in order to better motivate the discussion in later sections and in order to highlight the problems that we address in this chapter. We consider both cases of infinite precision and finite-precision arithmetic for reasons explained below.

6.1.1 The Drift Problem in Infinite-Precision Arithmetic

We illustrate the drift problem of LMS as follows. Consider the following contrived (deterministic) example. Let the regressors be scalar ($M = 1$) and given by $\boldsymbol{x}(k) = \frac{1}{\sqrt{k+1}}$. Assume also that the step-size μ is 1, that the noise sequence is $v(k) = 10^{-4}$, and that the "true" weight vector is $\boldsymbol{w}_* = 0$. It then follows from the model (6.2) and from the LMS recursion (1.6) that

$$\boldsymbol{w}_{k+1} = \left(1 - \frac{1}{k+1}\right)\boldsymbol{w}_k + 10^{-4}\frac{1}{\sqrt{k+1}}. \tag{6.1}$$

Solving this time-variant linear equation, we find that for a zero initial condition $\boldsymbol{w}_0$,

$$\boldsymbol{w}_k = \frac{10^{-4}}{k}\sum_{i=1}^{k}\sqrt{i} \;\geq\; \int_0^k \sqrt{\lambda}\, d\lambda = \frac{2\times 10^{-4}}{3}\sqrt{k}, \quad \text{for } k \geq 1,$$

which implies that $\boldsymbol{w}_k \to \infty$ as $k \to \infty$.

This example shows that the weight estimates computed by the LMS algorithm can grow slowly to very large values, even when the noise is small. Even with zero noise, unbounded growth of the estimates can happen due to finite-precision arithmetic errors (see [Cio87, CW85] and also the example below). Such unbounded growth of the LMS estimates may happen if two conditions are satisfied:

1. The noise or the finite-precision arithmetic errors have nonzero mean;
2. The covariance matrix of the input sequence $\{\boldsymbol{x}_k\}$ is not uniformly positive definite (i.e., there is no $\rho > 0$ such that $R_k > \rho I$ for all k).

As shown in [GMW82, CW85], these situations do arise in practice. For example, applications such as adaptive equalization with fractionally-spaced equalizers do not have inputs with uniformly positive-definite covariance matrices (see Sec. 2.2).

6.1.2 The Drift Problem in Fixed-Point Arithmetic

The example in this section shows how finite-precision errors can also cause drift. For this purpose, we assume that fixed-point arithmetic is used and employ the symbol fx$[a]$ to denote the fixed-point representation of a real number a. A fixed-point number is of the form

$$\pm 0.b_1 b_2 \ldots b_B,$$

where $b_i \in \{0, 1\}$. Note that this representation allows only for numbers that are less than 1 in magnitude:

$$|a| < 1.$$

Therefore, additions and subtractions may cause overflow, if the result lies outside the above range. Note also that the multiplication of two fixed-point number *never* causes overflow.

We denote by ε the *machine precision*, i.e., the largest absolute difference between a real number a and its fixed-point representation, namely $\left|\mathrm{fx}[a]-a\right| \leq \varepsilon$. For simplicity, we assume that all variables are stored with B bits plus sign, and that rounding is used (this implies that $\varepsilon = 2^{-B-1}$).

Finite-precision errors can result in nonzero mean variables in a number of ways. Consider, for example, a random variable a with distribution

$$a = \begin{cases} 0.5 + 2^{-7}, & \text{with probability } 0.5 - 2^{-7}, \\ -0.5 + 2^{-7}, & \text{with probability } 0.5 + 2^{-7}. \end{cases}$$

The expected value of a is $\mathrm{E}\, a = 0$. Assume, however, that a is quantized to fixed-point, with $B = 6$ bits plus sign (so that $\varepsilon = 2^{-7}$). If rounding is used, the quantized variable will have the distribution[1]

$$\mathrm{fx}[a] = \begin{cases} 0.5 + 2^{-6}, & \text{with probability } 0.5 - 2^{-7}, \\ -0.5, & \text{with probability } 0.5 + 2^{-7}\ . \end{cases}$$

The mean of $\mathrm{fx}[a]$ is -2^{-13}.

Another situation where finite-precision errors introduce nonzero mean variables is discussed in [CW85]. This reference shows that the rounding error of a product $\delta = \mathrm{fx}[a \cdot b] - ab$ may not have zero mean. The nonzero mean is caused by a non-symmetric implementation of the rounding function in arithmetic units that use two's complement arithmetic.

[1]The result depends on exactly how the rounding function is implemented. For example, $-0.5 + 2^{-7}$ might be rounded to $-0.5 + 2^{-6}$ in some machines.

For example, assume that we are using $B = 2$, and that the *exact* result of a multiplication is, for example, $a_1 = 1/2 + 1/16$, i.e.,

$$a_1 = 0.10\boxed{01} \text{ (in binary).}$$

Then, rounding the result to $B = 2$ bits plus sign simply discards (in this case) the last two bits, yielding

$$\text{fx}[a_1] = 0.10.$$

This is the *closest* fixed-point number to $1/8 + 1/16$.

On the other hand, if $a_2 = 1/2 + 1/8$, i.e.,

$$a_2 = 0.10\boxed{10} \text{ (in binary),}$$

there are *two equally close* fixed-point numbers to a_2, viz., $1/2$ and $1/2 + 1/4$. Which one is chosen depends on the implementation. We assume here that $1/2 + 1/4$ is chosen, i.e.,

$$\text{fx}[a_2] = 0.11.$$

Similarly, a negative number $a_3 = -1/2 - 1/8$ has two closest fixed-point representations, $-1/2$ and $-1/2 - 1/4$. In some implementations (normally ones that use two's complement arithmetic), the easier choice to implement in this case would be to choose $\text{fx}[a_3] = -1/2$. Therefore, we have

$$\text{fx}[1/2 + 1/8] - (1/2 + 1/8) = +1/8,$$

and

$$\text{fx}[-1/2 - 1/8] - (-1/2 - 1/8) = +1/8.$$

As we see, the rounding error of a positive number is positive, and the rounding error of the corresponding negative number is *also* positive. This causes the error

$$\delta = \mathrm{fx}[a.b] - a.b$$

to have a small, but nonzero mean. As explained in [CW85], although this problem may be avoided by proper design of the rounding functions, the solution increases a little the complexity of the arithmetic unit, and there are still several arithmetic processors that do have this problem.

Thus, a zero-mean variable may become nonzero-mean after quantization or after a fixed-point multiplication. This small nonzero mean might cause a slow drift of the LMS estimates, causing the algorithm to overflow. We illustrate this effect by simulating an $M = 2$ LMS filter whose input regressors satisfy (the values shown below are chosen such that the weight drift effect is amplified)

$$\boldsymbol{x}_k = \begin{cases} \begin{bmatrix} 0.5 & -0.5 \end{bmatrix}^T, & \text{with probability } 0.5, \\ \\ -\begin{bmatrix} 0.5 & -0.5 \end{bmatrix}^T, & \text{with probability } 0.5. \end{cases}$$

The noise is uniformly distributed with variance $\sigma_v^2 = 1/3 \times 10^{-3}$, the step-size is $\mu = 0.15$, and the true weight vector is $\boldsymbol{w}_* = \begin{bmatrix} \sqrt{0.2} & -\sqrt{0.2} \end{bmatrix}^T$. The arithmetic is again performed with $B = 6$ bits plus sign.

The weight estimates of the LMS recursion in finite precision are denoted by $\boldsymbol{z}_k$ and they are computed via (the rounding function is implemented as described in [CW85]):

$$\boldsymbol{z}_{k+1} \;=\; \boldsymbol{z}_k \;+\; \mathrm{fx}[\mu \boldsymbol{x}_k \, \mathrm{fx}[e(k)]] \; .$$

Figure 6.1 plots the values of $\|\boldsymbol{z}_k\|_\infty$. We see that overflow occurs at approximately $k = 250$ (the simulations in this section were computed using the random number generator from [PTV94]).

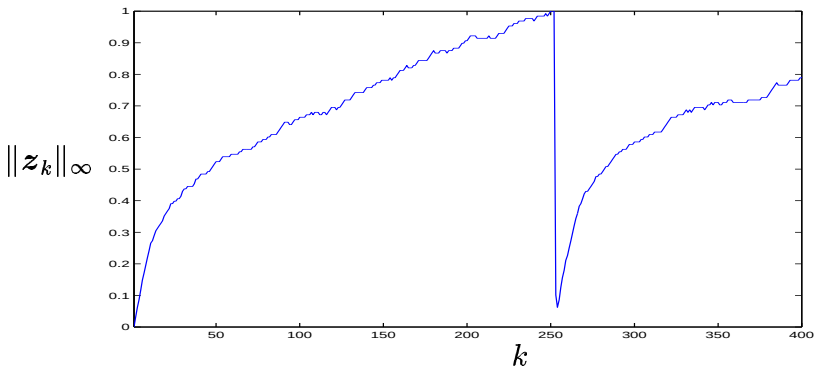

Figure 6.1: *Effect of small nonzero mean finite-precision error with the LMS algorithm. The plot shows* $\|\boldsymbol{z}_k\|_\infty$ *for the* $M = 2$ *example described in the text.*

6.2 OBJECTIVES

We saw in Sec. 2.1, Thms. 2.2 and 2.3 that, under certain independence assumptions, the LMS algorithm converges to steady-state values for the MSE and MSD. In this chapter, however, we are interested in studying the performance of LMS without any such assumptions. In particular, we would like to know if LMS or some variation thereof can always guarantee bounded estimates even in a worst-case scenario and in the presence of finite-precision effects. For this purpose, we shall see that:

i) LMS guarantees bounded weight estimates as long as the regressors $\{\boldsymbol{x}_k\}$ satisfy a certain PE condition. This result is well-known [IK84, SLJ86, SK95], and the PE condition is the natural equivalent of the requirement

$R > 0$ we had before in Assumption R-1 in the stochastic context. We shall, however, derive a bound for the worst-case rate of convergence of LMS (providing a different proof for another known result [WM79]).

ii) In finite-precision (when floating-point arithmetic is used), on the other hand, we show that the notion of PE regressors needs to be modified (see (6.16)) to require the regressors to be sufficiently PE in order to counter the effects of rounding errors. This expands results in [WM79], which considered only fixed-point arithmetic.

iii) When the PE assumptions do not hold, leaky LMS can still guarantee bounded estimates at the cost of bias and increased computations. This is also well-known.

iv) A new variant of LMS will be developed in Sec. 6.5.3 that will

   a) guarantee bounded estimates regardless of PE assumptions.

   b) not result in biased estimates.

   c) have the same effective cost as LMS.

We discuss all these issues in the sequel, starting with item i).

6.3 LMS IN INFINITE-PRECISION ARITHMETIC

We show in this section that LMS computes bounded estimates (i.e., drift does not occur) if a so-called *persistence of excitation* condition is satisfied. We shall also compute a worst-case bound for the rate of convergence of $\|\tilde{\boldsymbol{w}}_k\|$, providing a new proof for a result first published in [WM79]. In the next section, we also extend these results to finite-precision arithmetic.

In the deterministic setting, it is assumed that the scalar $y(k)$ and the length-M vector $\boldsymbol{x}_k$ are related through[2]

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k), \tag{6.2}$$

All variables are assumed to be bounded, i.e.,

D-1. *There exist constants B_x, B_v such that*

$$B_x = \sup_{k \geq 0} \boldsymbol{x}_k^T \boldsymbol{x}_k < \infty, \qquad B_v = \sup_{k \geq 0} |v(k)| < \infty. \tag{6.3}$$

The requirement of bounded $\{\boldsymbol{x}_k, v(k)\}$ is a standard one in the literature whenever finite-precision arithmetic effects are being studied, although it is often implicit in the assumptions. For example, the assumption that all variables are suitably scaled so that overflow never occurs in fact requires that all variables be bounded (see Appendix 2.2.B and the references [Cio87], [BB96a], [CL84], and [Ale87] for example).

6.3.1 Persistence of Excitation

The analog of the requirement $R > 0$ in the deterministic setting is the persistence of excitation (PE) condition . The condition is the following (compare with R-1 in Sec. 1.1).

Definition 2 (Persistence of excitation). *The bounded sequence $\{\boldsymbol{x}_k\}$ is said to be persistently exciting (PE) if there exist constants N and $a(N)$ such that the following relation is true for every $k_0 \geq 0$:*

$$0 < a(N) = \inf_{k_0 \geq 0} \lambda_{\min} \left(\sum_{k=k_0}^{k_0+N-1} \mu \boldsymbol{x}_k \boldsymbol{x}_k^T \right), \tag{6.4}$$

[2] Although we use the symbol $\boldsymbol{w}_*$ here, this is not necessarily the Wiener solution anymore (see the remark at the end of Sec. 1.1). Indeed, in most of this chapter, it is not assumed that $\boldsymbol{x}_k$ and $v(k)$ are uncorrelated (which of course would be the case if $\boldsymbol{w}_*$ were the Wiener solution).

where $\lambda_{\min}(A)$ is the smallest eigenvalue of the matrix A.

That is, the regressor sequence is persistently exciting if there exists a window of length N such that the above sum results positive definite for all values of $k_0 \geq 0$.

$\diamondsuit$

Note that for a given PE sequence, many choices for N and a are possible. In particular, if N_0 is one choice, then any $N > N_0$ is also a valid choice. The resulting a will generally be different for different N's, hence, a is a function of N, and we write $a(N)$.

6.3.2 Exponential Stability

Assume that the noise in (6.2) is identically zero, i.e., $v(k) \equiv 0$. Then, it is well-known that the origin $\tilde{\boldsymbol{w}} = \mathbf{0}$ of the error equation below is an exponentially stable equilibrium point [SK95, pp. 142–144].

$$\tilde{\boldsymbol{w}}_{k+1} = \left(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\right) \tilde{\boldsymbol{w}}_k, \quad \text{with initial condition } \tilde{\boldsymbol{w}}_0. \tag{6.5}$$

In the next lemma we provide a proof of this fact, as well as a bound for the worst-case rate of convergence. This bound depends on the notion of *level of excitation*, defined now.

Definition 3 (Level of excitation). *For every P such that $a(P) > 0$, define*

$$\gamma(P) \triangleq \frac{2 - \mu B_x}{\left(1 + \sqrt{P}\mu B_x\right)^2} \frac{a(P)}{\cdot} \tag{6.6}$$

The level of excitation of a PE sequence $\{\boldsymbol{x}_k\}$ satisfying $\mu B_x < 2$ is defined by

$$\gamma_0 \triangleq \sup_{a(P)>0} \gamma(P). \tag{6.7}$$

If sequence $\{\boldsymbol{x}_k\}$ *is not PE, we define* $\gamma_0 = 0$.

$\diamondsuit$

We will show in Lemma 6.1 that the rate of convergence for LMS is no worse (slower) than

$$\frac{\ln\big(1 - P\gamma(P)\big)}{P}.$$

That is, $\gamma(P)$ allows us to compute the worst-case rate of convergence for LMS. For floating-point arithmetic implementations, we will show (in Theorem 6.1 further ahead) that the machine precision must be compared with γ_0 to determine if a sequence $\{\boldsymbol{x}_k\}$ will result in a guaranteed-stable LMS filter. In other words, we will show in Thm. 6.1 that, the smaller γ_0 is, the more precise the floating-point computations must be for LMS to remain stable (this is a new result).

Note that there may not be a value of N such that $\gamma(N) = \gamma_0$. Nevertheless, for every $\delta > 0$ there is a finite N such that $\gamma_0 - \delta < \gamma(N) \leq \gamma_0$ (which follows directly from the definition of sup).

Lemma 6.1. *Assume that the noise is identically zero, that the sequence* $\{\boldsymbol{x}_k\}$ *is bounded as in* (6.3) *with* $\mu B_x < 2$ *and that it is also PE with level*

$$\gamma_0 > 0. \tag{6.8}$$

Let $N > 0$ *be such that* $0 < \gamma(N) \leq \gamma_0$. *Under these conditions the following inequality holds*

$$\|\tilde{\boldsymbol{w}}_{k+N}\|^2 \leq \big(1 - N\gamma(N)\big)\,\|\tilde{\boldsymbol{w}}_k\|^2. \tag{6.9}$$

Hence, the origin $\tilde{\boldsymbol{w}} = \boldsymbol{0}$ *in* (6.5) *is exponentially stable, and the worst-case rate of convergence is*

$$\lambda \triangleq e^{\inf_{a(N)>0} \frac{\log\left(1 - N\gamma(N)\right)}{N}}.$$

If the noise sequence is nonzero, but bounded as in Assumption D-1, then under the conditions of this lemma the weight-error vector $\tilde{\boldsymbol{w}}_k$ is also bounded

$$\sup_{k\geq 0} \|\tilde{\boldsymbol{w}}_k\| < \infty.$$

Proof: See Appendix 6.B.

$\diamondsuit$

Compare the condition $\mu B_x < 2$ with the bound (4.32) from Chapter 4 and Fig. 4.2. Although the problems considered here and in Chapter 4 are essentially different (one is stochastic and the other deterministic), Lemma 6.1 hints that a tight bound for the behavior of $\mu_{\max}$ should eventually be inversely proportional to the filter length M, since (in the notation of Chapter 4) if

$$\boldsymbol{x}_k = \Big[a(k-M+1) \quad a(k-M+2) \quad \ldots \quad a(k)\Big]^T,$$

and $|a(k)| < A < \infty$ for all k, then $B_x < MA^2$, and the condition $\mu B_x < 2$ for (deterministic) stability becomes

$$\mu < \frac{2}{MA^2}.$$

Note, though, that Lemma 6.1 does not apply to important *unbounded* input sequences (such as Gaussian-distributed sequences).

6.4 LMS IN FINITE-PRECISION ARITHMETIC

This section studies the stability of the finite-precision LMS. The effects of a non-zero noise sequence $\{v(k)\}$ will also be considered. We present results only for floating-point arithmetic; results for fixed-point (which are in fact simpler)

will be presented in Sec. 6.5.2, when we discuss leakage-based algorithms, and were also discussed in [WM79] for the special case of LMS.

A floating-point number is a quantity of the form [GL89, p. 61]

$$\pm 0.b_1 b_2 \ldots b_t \times 2^e$$

where $b_i \in \{0, 1\}$, and t is the number of bits in the representation. The quantity $b = 0.b_1 b_2 \ldots b_t$ is the *mantissa*, and $e_m \leq e \leq e_M$ is the *exponent* of the floating-point number (e_m and e_M are integers, normally $e_m < 0$ and $e_M > 0$). In general, the representation allows only either $b_1 \neq 0$, or all $b_i = 0$. Of course, not all real numbers can be represented in the above form.

Floating point additions, subtractions, and multiplications are (usually) performed with accuracy $\varepsilon \stackrel{\Delta}{=} 2^{-t}$, i.e.,

$$\mathrm{fl}(x \circ y) = (x \circ y)(1 + \delta), \tag{6.10}$$

where $|\delta| < \varepsilon$ and $\circ$ denotes $+$, $-$, or $\times$. This is true for floating point processors that adhere to the IEEE standards. The symbol δ will generally denote quantities that are bounded by multiples of the machine precision.

Quantization errors in rounding $\boldsymbol{x}_k$ and $y(k)$ to floating-point will not be considered in the analysis below; the dynamics of the algorithm is the primary concern. These additional quantization disturbances can be included in the analysis and their effect would be essentially to increase by one small multiple of ε the lower bound requirement on γ_0; see (6.16) further ahead.

In floating-point, exact quantities should be replaced by computed quantities. The computed weight estimate will be denoted by $\boldsymbol{z}_k$. Hence, the *computed* error $e(k)$ would be given by $\mathrm{fl}\big(y(k) - \boldsymbol{x}_k^T \boldsymbol{z}_k\big)$. It is known from rounding error analyses that ([GL89, pp. 63–64] and also [Ste73, Wil63]):

$$\mathrm{fl}\big(\boldsymbol{x}_k^T \boldsymbol{z}_k\big) = \boldsymbol{x}_k^T \boldsymbol{z}_k + \delta_1' . \|\boldsymbol{x}_k\| . \|\boldsymbol{z}_k\|, \tag{6.11}$$

where $|\delta_1'| < 1.01M\varepsilon$ (if $M\varepsilon < 0.01$). That is, the rounding error in the inner product computation $\boldsymbol{x}_k^T\boldsymbol{z}_k$ is dependent on the norms of the vectors $\boldsymbol{x}_k$ and $\boldsymbol{z}_k$. Using (6.11) it follows that

$$\mathrm{fl}\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{z}_k\big) = \Big(y(k) - \boldsymbol{x}_k^T\boldsymbol{z}_k + \delta_1'\|\boldsymbol{x}_k\|\,\|\boldsymbol{z}_k\|\Big)(1+\delta_2'),$$

where $|\delta_2'| < \varepsilon$. Define $\check{e}(k) \triangleq y(k) - \boldsymbol{x}_k^T\boldsymbol{z}_k$ to denote the *infinite precision* value of the error when $\boldsymbol{z}_k$ is used (this is similar to $e(k)$ in infinite-precision, with $\boldsymbol{z}_k$ replacing $\boldsymbol{w}_k$). Then

$$\mathrm{fl}\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{z}_k\big) = \check{e}(k) + \delta_2'e(k) + \delta_1'(1+\delta_2')\|\boldsymbol{x}_k\|\|\boldsymbol{z}_k\|. \tag{6.12}$$

Define the weight error vector $\tilde{\boldsymbol{z}}_k = \boldsymbol{w}_* - \boldsymbol{z}_k$. It measures how far the computed weight estimate is from $\boldsymbol{w}_*$. The computed estimates $\boldsymbol{z}_k$ are propagated via:

$$\boldsymbol{z}_{k+1} = \mathrm{fl}\Big(\boldsymbol{z}_k + \mu\boldsymbol{x}_k\big(y(k) - \boldsymbol{x}_k^T\boldsymbol{z}_k\big)\Big), \tag{6.13}$$

which is the floating-point version of the LMS recursion (1.6). Using again standard rounding error results, the finite-precision error equation becomes:

$$\begin{aligned}\tilde{\boldsymbol{z}}_{k+1} = {} & \big[I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big]\,\tilde{\boldsymbol{z}}_k - \mu\boldsymbol{x}_k v(k) - \\ & - \mathrm{diag}\big(\delta_{1,l}\big)\boldsymbol{z}_k - \mathrm{diag}\big(\delta_{2,l}\big)\mu\check{e}(k)\boldsymbol{x}_k - \mathrm{diag}\big(\delta_{3,l}\big)\mu\|\boldsymbol{x}_k\|\|\boldsymbol{z}_k\|\boldsymbol{x}_k,\end{aligned} \tag{6.14}$$

where $\mathrm{diag}(\delta_{i,\,l})$ are $M\times M$ diagonal matrices with elements $\delta_{i,\,l}$, $l = 0\ldots M-1$, and $\big|\delta_{1,l}\big| < \varepsilon \triangleq \delta_1$, $\big|\delta_{2,l}\big| < 4.04\varepsilon \triangleq \delta_2$ and $\big|\delta_{3,l}\big| < 1.02M\varepsilon \triangleq \delta_3$.

In infinite-precision this equation would have been

$$\tilde{\boldsymbol{w}}_{k+1} = \big(I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big)\tilde{\boldsymbol{w}}_k - \mu\boldsymbol{x}_k v(k). \tag{6.15}$$

The goal of the current analysis is to derive conditions under which, in finite precision, the sequence $\big\{\tilde{\boldsymbol{z}}_k\big\}$ in (6.14) is bounded, and to show that in steady-state, $\|\tilde{\boldsymbol{z}}_k\|^2$ will be bounded by a weighted sum of B_v^2 and ε. These results are

useful since they provide conditions that guarantee a stable performance of LMS even in finite precision.

It will follow as a result of this analysis that the PE definition (6.4) will need to be strengthened so as to require that the regressor sequence be *sufficiently* PE. This is because in floating-point arithmetic, it is not enough to require $a(N)$ to be positive. It has to be sufficiently away from zero by an amount that depends on ε, the machine precision (see (6.16)). The derivations to establish these facts are relatively lengthy, for this reason the derivations will be deferred to Appendix 6.C.

Theorem 6.1 (Floating-point stability of LMS). *If the sequence* $\{\boldsymbol{x}_k\}$ *is PE with a level of excitation* γ_0 *satisfying*

$$\gamma_0 > \xi_2, \tag{6.16}$$

then for any $N \geq M$, *such that* $\gamma(N) \in (\xi_2,\ \gamma_0]$, $\|\tilde{\boldsymbol{z}}_k\|^2$ *will be bounded by the equation below if* k *is large.*

$$\|\tilde{\boldsymbol{z}}_k\|^2 \leq C\frac{\left(\frac{1}{\rho}\xi_3 + \xi_4\right)\|\boldsymbol{w}_*\|^2 + \left(\frac{1}{\rho}\big(1+\xi_5\big) + 1 + \xi_6\right)\mu B_x B_v^2}{\gamma(N) - \rho\big(1+\xi_1\big) - \xi_2}, \tag{6.17}$$

where the ξ_i *are constants dependent on* ε, M *and* μB_x. *The constant* C *is computed in Appendix 6.C, and is equal to* N *for large* μB_x, *and approximately* 1 *if* $\mu N B_x \ll 1$. *The positive number* ρ *is small enough so that* $\gamma(N) > \xi_2 + \rho(1+\xi_1)$.

The worst-case rate of convergence for floating-point implementations of LMS is:

$$\lambda_{FP} \stackrel{\Delta}{=} e^{\inf_{\gamma(N)>\xi_2} \frac{1-N\gamma(N)+\xi_2}{N}}. \tag{6.18}$$

For small ε and moderate values of M, the ξ_i are approximately

$$\xi_1 \leq \big(2 + 8.1\mu B_x\big)\varepsilon \qquad \xi_2 \leq \big(6.1 + 4.1\beta + 2.1\mu B_x M\big)\varepsilon \quad \xi_3 \leq \big(1 + 1.1\mu B_x M\big)\varepsilon$$

$$\xi_4 \leq \big(1 + 1.1\mu B_x M\big)\varepsilon \quad \xi_5 \leq \big(5.1 + 4.1\mu B_x + 1.1\beta M\big)\varepsilon \quad \xi_6 \leq \big(9.08 + 1.1\mu B_x M\big)\varepsilon$$

Proof: See Appendix 6.C.

$\diamondsuit$

Expression (6.17) essentially says that for large enough k, $\|\tilde{\boldsymbol{z}}_k\|^2$ tends to a ball whose radius is approximately determined by

$$\frac{O(\varepsilon)}{\gamma}\big(\|\boldsymbol{w}_*\|^2 + \mu B_x B_v^2\big) + \frac{\text{Constant}}{\gamma}\mu B_x B_v^2.$$

Condition (6.16) says that the weight error vector sequence $\{\tilde{\boldsymbol{z}}_k\}$ will be bounded if the regressor sequence excites all directions in $\mathbb{R}^M$ with a level larger than a multiple of the machine precision ε.

Relation (6.17) is a worst-case bound. It is not intended to be a tight bound on the performance of the floating-point LMS.[3] Its usefulness is to show that the estimate errors $\{\tilde{\boldsymbol{z}}_k\}$ obtained through FP LMS form a bounded sequence if the PE condition (6.16) is satisfied. It also shows that if the unit round-off ε and the step-size μ are reduced, the steady-state error can be made as small as one wishes. Finally, it shows that if ε is kept constant, the step-sizes can be reduced only as long as $\gamma_0 > \xi_2$ (recall that γ_0 depends linearly on the step-sizes).

The analysis for the LMS algorithm implemented in fixed-point arithmetic is simpler, and will be presented in parallel with our discussion of the drift problem and of leakage-based algorithms in the next sections. Reference [WM79] analyzes

[3] From now on, a floating-point (fixed-point) implementation of an algorithm will be referred to by using the prefix "FP" ("FX") — for example, "FP LMS", "FP leaky LMS", ("FX LMS", "FX leaky LMS") etc. For the theoretical, infinite-precision versions, the prefix "IP" is used.

fixed-point implementations of the LMS algorithm (the reference claims that their results hold for an implementation mixing floating and fixed-point arithmetic, but the error model in [WM79] does not include some important features of the floating-point LMS error equation, including the error proportional to $\|\boldsymbol{z}_k\|$ in (6.12)).

6.5 LEAKAGE-BASED ALGORITHMS

When the above persistence of excitation conditions (6.8) and (6.16) are not satisfied, the LMS algorithm can become unstable (as we saw in Sec. 6.1). The leaky LMS algorithm was proposed in the early seventies to avoid this problem [Zah73, CM75, GMW82]. Leaky LMS, however, introduces a bias problem that was also described and analyzed in these references, as well as in [MA97] and others (see Sec 2.3).

6.5.1 Solution of Drift Problem by Leakage

The leakage term in (6.19) below prevents unbounded growth of the weight vectors from occurring:

$$\boldsymbol{w}_{k+1}^l = \big(1 - \mu\alpha_0\big)\boldsymbol{w}_k^l + \mu\boldsymbol{x}_k e^l(k), \tag{6.19}$$

where we use the symbol $e^l(k)$ to indicate that the output error is computed using the leaky LMS estimate $\boldsymbol{w}_k^l$.

In the first of our examples (Sec. 6.1.1), using leakage we obtain the following recursion for the error vector

$$\tilde{\boldsymbol{w}}_{k+1}^l = \left(1 - \mu\alpha_0 - \frac{\mu}{k+1}\right)\tilde{\boldsymbol{w}}_k^l + \mu\alpha_0\boldsymbol{w}_* - \mu\frac{10^{-4}}{\sqrt{k+1}}\,.$$

This recursion results in a bounded sequence $\{\tilde{\boldsymbol{w}}_k^l\}$ if $0 < \mu < 2/(\alpha_0 + 1)$. More

generally, the following fact can be established for leaky LMS (see [SLJ86] — this known result also follows from a simple modification of our stability proof for switching-σ in Thm. 6.4).

Lemma 6.2 (BIBS Stability of Leaky LMS). *Consider the leaky LMS algorithm (6.19) in infinite-precision arithmetic. If $\mu < 2/(\alpha_0 + B_x)$, then $\|\boldsymbol{w}_k^l\|$ remains bounded if the noise sequence $\{v(k)\}$ is bounded.*

$\diamondsuit$

In other words, under the condition $\mu < 2/(\alpha_0 + B_x)$, the leaky LMS algorithm is bounded-input bounded-state (BIBS)-stable, where we treat the weight estimates as the state and the noise sequence $v(k)$ as the input. This result can be extended to finite-precision arithmetic, as will follow from the arguments we provide in Sec. 6.7. The lemma below is a new result.

Lemma 6.3 (Fixed-point Stability of Leaky LMS). *The leaky LMS algorithm implemented in fixed-point arithmetic guarantees that the sequence $\{\boldsymbol{w}_k^l\}$ is bounded if*

$$|1 - \mu\alpha_0 - \mu B_x| \leq |1 - \mu\alpha_0| < 1.$$

and if $\{v(k)\}$ is bounded.

Proof: This result follows from the proof of Thm. 6.4 further ahead.

$\diamondsuit$

6.5.2 The Bias Problem of Leaky LMS

Although the leaky LMS algorithm solves the weight-drift problem, it leads to biased estimates, as we saw in Sec. 2.3. Indeed, if the step-size and leakage parameters are properly chosen we obtain, assuming now stochastic variables,

$$\lim_{k\to\infty} \mathrm{E}\,\tilde{\boldsymbol{w}}_k^l = \alpha_0\big(\alpha_0 I + R\big)^{-1}\boldsymbol{w}_*. \tag{6.20}$$

That is, the average weight error $\mathrm{E}\,\tilde{\boldsymbol{w}}_k^l$ computed by the leaky LMS algorithm will not converge to zero, even in ideal conditions (positive-definite R, zero noise, and no quantization errors) — recall the discussion in Sec. 2.3.

The conventional solution to the bias problem in (6.19) has been to use a very small α_0. However, this choice has its drawbacks. A value of α_0 too small might not be capable of countering the effects of finite-precision arithmetic. In addition, even a small α_0 might create a significant bias, as shown in the simulation in Figure 6.2. The lighter curve is the plot of the squared error $e^l(k)^2$ (not its average), computed by the leaky LMS algorithm for the same environment as in Fig. 6.1 (same $\boldsymbol{w}_*$ and noise and input statistics). The step-size is again $\mu = 0.15$, and the leakage parameter is $\mu\alpha_0 = 2^{-6}$. Note that this is the second smallest value that could be chosen for $\mu\alpha_0$, corresponding to twice the value of the least-significant bit (1 LSB $= 2^{-7}$ in this example).

The darker curve is a plot of the squared error computed by the LMS algorithm, $e(k)^2$. Almost all the time, $e(k)^2$ is smaller than $e^l(k)^2$, but there are spikes when overflow occurs. [This kind of sudden worsening of the performance is what turns the LMS filter unusable for some applications.] Comparing the results for LMS and leaky LMS, we note that although the latter avoids overflow, the level of the error is significantly increased. In addition, recall from Table 6.1 that leaky LMS has a computational cost higher than that of LMS. More examples

are provided in Sec. 6.8.

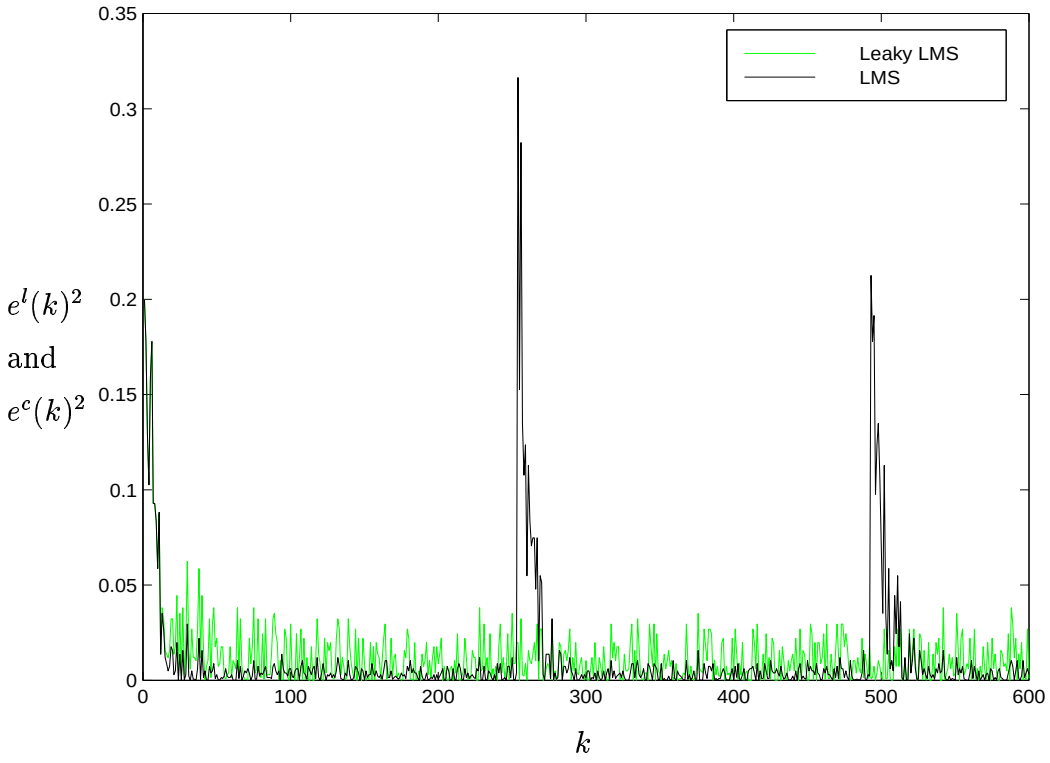

Figure 6.2: *Comparison of the squared errors $e(k)^2$ (LMS) and $e^l(k)^2$ (leaky LMS) for the $M = 2$ example of Fig. 6.1. The darker curve with the spikes corresponds to LMS. No average was performed.*

6.5.3 A New Leakage-Based Algorithm

The above examples and discussion motivate us to pursue other ways to solve the bias and drift problems of LMS and leaky LMS without compromising the performance of the adaptive algorithm. We do so by introducing a new leakage-based algorithm, which we shall call *circular-leaky*, and also by studying the performance of a modified switching-σ algorithm (6.24).

Circular-leaky. In this new algorithm, we employ a nonlinear and time-variant leakage term $\alpha_c(k, \cdot)$ instead of the constant factor α_0 in leaky LMS (6.19). We denote the resulting weight vector estimate by $\boldsymbol{w}_k^c$ and its taps (or entries) by $w_{k,j}^c$, for $j = 1, \ldots, M$. There are two modifications with respect to leaky LMS in this new variant. First, leakage is applied to a single tap at each iteration; and secondly, leakage is applied only if the tap magnitude exceeds a pre-specified level, say C_1.

More specifically, at time k we first compute $e^c(k) = y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k^c$, and then check whether $|w_{k,\bar{k}}^c| > C_1$, where

$$\bar{k} \stackrel{\Delta}{=} (k \mod M).$$

If the condition is true, we compute an intermediate estimate, $\bar{\boldsymbol{w}}_k^c$, that is identical to $\boldsymbol{w}_k^c$ except for a leakage term that is applied to its $\bar{k}-$th entry, as shown below

$$\bar{\boldsymbol{w}}_k^c = \begin{cases} \left[w_{k,0}^c \quad \ldots \quad \big(1 - \mu\alpha_c(k,\, w_{k,\bar{k}}^c)\big) w_{k,\bar{k}}^c \quad \ldots \quad w_{k,M-1}^c \right]^T, & \text{if } \left|w_{k,\bar{k}}^c\right| > C_1, \\ \boldsymbol{w}_k^c\,, & \text{otherwise.} \end{cases}$$

Note that at most one entry of $\boldsymbol{w}_k^c$ is modified in the computation of $\bar{\boldsymbol{w}}_k^c$ (the value of the leakage term $\alpha_c(k,\, w_{k,\bar{k}}^c)$ is defined further ahead). Once the intermediate estimate $\bar{\boldsymbol{w}}_k^c$ has been computed, we proceed with an LMS-type update, namely

$$\boldsymbol{w}_{k+1}^c = \bar{\boldsymbol{w}}_k^c + \mu \boldsymbol{x}_k e^c(k). \tag{6.21}$$

We can describe the algorithm more compactly as follows. Let $\boldsymbol{e}_{\bar{k}}$ denote the $\bar{k}$-th basis vector (i.e., $e_{\bar{k},\bar{k}} = 1$, $e_{\bar{k},j} = 0$ for $j \neq \bar{k}$). Then the new algorithm takes the

form:

$$\boldsymbol{w}_{k+1}^c = \left(I - \mu\alpha_c(k,\, w_{k,\,\bar{k}}^c)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\right)\boldsymbol{w}_k^c + \mu\boldsymbol{x}_k e^c(k)\ . \tag{6.22}$$

The function $\alpha_c(k,\,\cdot)$ is defined as follows. Let α_0, C_1, and $C_2 > C_1$ be positive constants, and define $D = (C_2 - C_1)/2$. Then[4]

$$\alpha_c(k,\, w_{k,\,\bar{k}}^c) = \begin{cases} \alpha_0 & \text{if } \left|w_{k,\,\bar{k}}^c\right| \geq C_2, \\ \alpha_0 - \frac{\alpha_0}{2}\left(\frac{C_2 - |w_{k,\,\bar{k}}^c|}{D}\right)^2 & \text{if } C_1 + D \leq \left|w_{k,\,\bar{k}}^c\right| < C_2, \\ \frac{\alpha_0}{2}\left(\frac{|w_{k,\,\bar{k}}^c| - C_1}{D}\right)^2 & \text{if } C_1 < \left|w_{k,\,\bar{k}}^c\right| < C_1 + D, \\ 0 & \text{otherwise.} \end{cases} \tag{6.23}$$

In words, starting from $k = 0$, we examine the magnitude of the *top* entry of $\boldsymbol{w}_0^c$ and check in which interval it lies,

$$(0, C_1], \qquad (C_1, C_1 + D), \qquad [C_1 + D, C_2), \qquad (C_2, \infty).$$

The interval tells us the value of the leakage, α_c, that we should apply to this tap entry. In this way we create $\bar{\boldsymbol{w}}_0^c$ and then $\boldsymbol{w}_1^c$ via (6.21).

Next, we examine the magnitude of the *second* entry of $\boldsymbol{w}_1^c$, determine in which interval it lies, and find the appropriate α_c. We then generate $\bar{\boldsymbol{w}}_1^c$ and $\boldsymbol{w}_2^c$ via (6.21).

Next, we examine the magnitude of the *third* entry of $\boldsymbol{w}_2^c$, determine α_c, $\bar{\boldsymbol{w}}_2^c$, and $\boldsymbol{w}_3^c$. We continue in this fashion by examining in each iteration k a single

[4]It is possible to simplify this definition and use a discontinuous $\alpha_c(k,\, w_{k,\,\bar{k}}^c)$ – see Sec. 6.9 and also [NS96].

entry of $\boldsymbol{w}_k^c$, and by moving circularly from one entry to another as the iterations progress.

The time dependency of α_c comes from the fact that a different entry of $\boldsymbol{w}_k^c$ is checked at each time instant. To simplify the notation, we will not explicitly indicate this time dependency in the remainder of the chapter, and will thus write $\alpha_c(w_{k,\bar{k}}^c)$ instead of $\alpha_c(k,\, w_{k,\bar{k}}^c)$. In Fig. 6.3 we plot $\alpha_c(\cdot)$ for $C_1 = 0.5$, $C_2 = 0.7$, and $\alpha_0 = 0.1$. Later in the chapter (see, e.g., Eq. (6.31)) we show how $\{\mu, \alpha_0, C_1\}$ should be chosen. For now, note only that C_1 must satisfy $C_1 > \|\boldsymbol{w}_*\|_\infty$ in order to guarantee that the leakage term $\alpha_c(\cdot)$ is zero when the estimate $\boldsymbol{w}_k$ is close to $\boldsymbol{w}_*$. In the following we assume that a bound $W_\infty \geq \|\boldsymbol{w}_*\|_\infty$ is available.

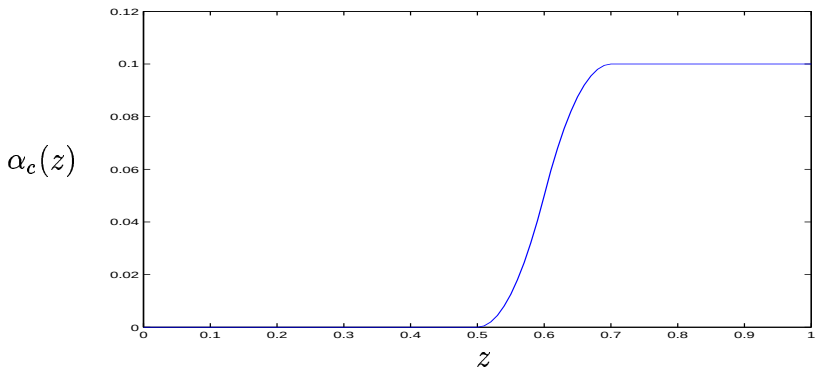

Figure 6.3: *Function* $\alpha_c(\cdot)$ *with* $C_1 = 0.5$, $C_2 = 0.7$, *and* $\alpha_0 = 0.1$.

Modified switching-σ algorithm. In this algorithm, the leakage factor is applied to all taps whenever $\|\boldsymbol{w}_k^s\|$ is too large ($\|\cdot\|$ represents the Euclidean norm),

$$\boldsymbol{w}_{k+1}^s = \big(1 - \mu\alpha_s(\boldsymbol{w}_k^s)\big)\boldsymbol{w}_k^s + \mu\boldsymbol{x}_k e^s(k), \tag{6.24}$$

where the function $\alpha_s(\boldsymbol{w}_k^s)$ is defined as follows. Let α_0, S_1, and $S_2 > S_1$ be

positive constants, and define $E = (S_2 - S_1)/2$. Then

$$\alpha_s(\boldsymbol{w}_k^s) = \begin{cases} \alpha_0 & \text{if } \|\boldsymbol{w}_k^s\| \geq S_2 \\ \alpha_0 - \frac{\alpha_0}{2}\left(\frac{S_2 - \|\boldsymbol{w}_k^s\|}{E}\right)^2 & \text{if } S_1 + E \leq \|\boldsymbol{w}_k^s\| < S_2, \\ \frac{\alpha_0}{2}\left(\frac{\|\boldsymbol{w}_k^s\| - S_1}{E}\right)^2 & \text{if } S_1 < \|\boldsymbol{w}_k^s\| < S_1 + E, \\ 0 & \text{otherwise.} \end{cases} \tag{6.25}$$

As was the case for circular-leaky, the constant S_1 is chosen so that leakage is off when $\boldsymbol{w}_k$ is close to $\boldsymbol{w}_*$ (i.e., we assume that a bound $W_2 \geq \|\boldsymbol{w}_*\|$ is available, and choose $S_1 > W_2$). A variant of this algorithm is well-known in the adaptive control literature [IT86], where the leakage function $\alpha_s(\boldsymbol{w}_k^s)$ is not smooth as above, but has a discontinuous (abrupt) transition between 0 and α_0.

Table 6.2 summarizes the different ways in which leakage is introduced in each algorithm.

Algorithm	Leakage applied if	Leakage term
Leaky LMS	always applied	$\mu\alpha_0\boldsymbol{w}_k^l$
Circular-leaky	$\left\|w_{k,\bar{k}}^c\right\| > C_1$	$\mu\alpha_c\left(w_{k,\bar{k}}^c\right)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_k^c$
Modified switching-σ	$\left\|\boldsymbol{w}_k^s\right\| > S_1$	$\mu\alpha_s\left(\boldsymbol{w}_k^s\right)\boldsymbol{w}_k^s$

Table 6.2: *Differences in the leakage terms among the algorithms.*

The purpose of the discussion in the sequel is twofold:

i) To establish that the modified switching-σ and circular-leaky algorithms solve the drift problem *even* under the more demanding environment of a finite-precision implementation. In particular, we determine conditions on the leakage parameters so that rounding effects will not contribute to drift.

ii) To establish that both algorithms also compute asymptotically unbiased estimates when the regressor covariance matrix is positive-definite ($R > 0$).

We employ two tools in our analysis. The first tool is a stochastic averaging analysis (see Sec 3.3), which will be used in Sec. 6.6 to establish point ii) above, namely that the modified switching-σ and circular-leaky algorithms compute asymptotically unbiased estimates. The second tool is based on a (deterministic) Lyapunov stability analysis, which will be used in Sec. 6.7 to show that both algorithms avoid unbounded growth of the weight error vector.

6.6 STOCHASTIC PERFORMANCE ANALYSIS

In this section we show that the estimates provided by circular-leaky and switching-σ algorithms are unbiased. In fact, we establish a stronger conclusion, namely that this property holds even when using fixed-point arithmetic with rounding. These results are established by relying on averaging methods, which we reviewed in Sec. 3.3.

We assume in this section that $\{y(k), \boldsymbol{x}_k\}$ are jointly stationary stochastic sequences, with $\mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T = R > 0$ and $\{\boldsymbol{x}_k\}$ uniform-mixing and bounded, i.e., we assume that:

M-1. *The sequences $\{y(k), \boldsymbol{x}_k\}$ are related via a linear model of the form*

$$y(k) = \boldsymbol{x}_k^T\boldsymbol{w}_* + v(k)$$

for some unknown $\boldsymbol{w}_*$, and where $v(k)$ is zero-mean with variance σ_v^2, and uncorrelated with $\boldsymbol{x}_k$, i.e.,

$$\mathrm{E}\,v(k)\boldsymbol{x}_k = \boldsymbol{0}.$$

IS-1. *The random sequences* $\{\boldsymbol{x}_k\}$ *and* $\{y(k)\}$ *are jointly stationary.*

R-1. *The matrix* R *is invertible.*

UM-1. *The sequence* $\{\boldsymbol{x}_k\}$ *is uniform mixing with mixing function* $\varphi(n)$.

B-1. *The sequence* $\{\boldsymbol{x}_k\}$ *satisfies*

$$\sup_{k>0} \|\boldsymbol{x}_k\|^2 \leq B_x < \frac{2}{\mu}, \quad \textit{with probability 1.}$$

Note that this last assumption is a stochastic version of (6.3).

6.6.1 Circular-Leaky Algorithm

We will analyze here a fixed-point implementation of the circular-leaky algorithm. As in Sec. 6.4, we need to distinguish between the infinite-precision and the finite precision versions of the update laws. For this reason, we shall denote the weight error vector in fixed-point by $\tilde{\boldsymbol{z}}_k^c$ (and reserve $\tilde{\boldsymbol{w}}_k^c$ for the infinite-precision case). Using Eq. (6.D.5) from Appendix 6.D, we can show that $\tilde{\boldsymbol{z}}_k^c$ satisfies the following recursion

$$\tilde{\boldsymbol{z}}_{k+1}^c = \left(I - \mu\alpha_c(z_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\right)\tilde{\boldsymbol{z}}_k^c + \mu\alpha_c(z_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_* - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\delta}_k^c \,, \tag{6.26}$$

where the variable $\boldsymbol{\delta}_k^c$ accounts for all finite-precision errors and satisfies

$$\|\boldsymbol{\delta}_k^c\| \leq \left(\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1(\alpha_c \neq 0)\right)\varepsilon \,, \tag{6.27}$$

$$\mathrm{E}\,\boldsymbol{\delta}_k^c\boldsymbol{\delta}_k^{c\,T} = \Sigma_\xi + \mu^2\sigma_d^2 MR + 1(\alpha_c \neq 0)\sigma_d^2\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T \,, \tag{6.28}$$

$$|\delta_{k,j}^c| \leq \left(2 + (1+\mu)\|\boldsymbol{x}_k\|_\infty\right)\varepsilon \,, \tag{6.29}$$

where $1(\alpha_c \neq 0) = 1$ if $\alpha_c \neq 0$ and zero otherwise,

$$\Sigma_\xi = \sigma_d^2\big(I + R\big),$$

and $z_{k,\bar{k}} = w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}$ with $\bar{k} = (k \bmod M)$.

To use Theorem 3.4, we need to prove that the fixed-point circular-leaky error equation (6.26) and its averaged counterparts satisfy the conditions (i)–(iii) given in the statement of the theorem. Averaging the error equation (6.26) over the input $\boldsymbol{x}_k$, the noise $v(k)$, and the finite-precision errors $\boldsymbol{\delta}_k^c$, we obtain the recursion[5]

$$\tilde{\boldsymbol{z}}_{k+1}^{av} = \big(I - \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R\big)\tilde{\boldsymbol{z}}_k^{av} + \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*. \quad (6.30)$$

It is shown in Thm 6.2 below that this recursion satisfies conditions (i)-(iii) for values of μ and α_0 that satisfy

$$\mu\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < 2 - \mu\lambda_{\max}(R)\,, \quad (6.31)$$

where $\eta_c > 0$ is a constant that satisfies $C_1 \geq (1+\eta_c)\|\boldsymbol{w}_*\|_\infty$. The partially averaged system is further given by

$$\tilde{\boldsymbol{z}}_{k+1}^{pav} = (I - \mu R)\tilde{\boldsymbol{z}}_k^{pav} - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\delta}_k^c.$$

This is in fact the same partially averaged recursion that would result for the LMS algorithm in fixed-point arithmetic. Therefore, in steady-state, the circular-leaky algorithm will behave like the LMS algorithm. In particular, circular-leaky computes asymptotically unbiased estimates, since the estimates computed by LMS have this property. The value of the steady-state error $\lim_{k\to}\mathrm{E}\,e^c(k)^2$ can then be obtained from Appendix 2.B in Chapter 2. We state the results below, with the necessary conditions.

Theorem 6.2 (Steady-state performance of circular-leaky). *Assume that* $\{\boldsymbol{x}_k\}$, $\{v(k)\}$, *and* $\{\boldsymbol{\delta}_k^c\}$ *are stationary, have zero mean, and satisfy* $\mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T =$

[5]To simplify the notation, in this section we will drop the superscript c from the averaged variables.

$R > 0$. *Assume further that* $\{v(k)\}$ *is iid and independent of* $\{\boldsymbol{x}_k\}$, *and that this last sequence is uniform-mixing (i.e., it satisfies Assumption UM-1). Then, if the step-size* μ *is small enough and condition* (6.31) *holds, the circular-leaky estimates* $\boldsymbol{z}_k^c$ *are asymptotically unbiased, and in steady-state we have*

$$\lim_{k\to\infty} \mathrm{E}\,\|\tilde{\boldsymbol{z}}_k\|^2 \approx \frac{\mu}{2}\left(\sigma_v^2 + M\sigma_d^2\right) M + \frac{\sigma_d^2}{2\mu}\,\mathrm{Tr}\left(I + R^{-1}\right). \tag{6.32}$$

$$\lim_{k\to\infty} \mathrm{E}\,e^c(k)^2 \approx \sigma_v^2 + \mu(\sigma_v^2 + \sigma_d^2)\frac{\mathrm{Tr}(R)}{2} + \frac{\mathrm{Tr}(\sigma_d^2(I+R))}{2}\,, \tag{6.33}$$

where $\sigma_d^2 = 2^{-2B}/12$ *for a fixed-point implementation with* B *bits plus sign (as shown in Appendix 6.D).*

Proof: The argument is lengthy. In order to apply Theorem 3.4, we need first to show that the fixed-point circular-leaky error equation (6.26), and its averaged counterparts, satisfy the conditions (i)–(iii) given in the statement of Thm. 3.4. Dropping the superscript c from the averaged variables for ease of notation, the averaged error equation becomes (cf. (6.30))

$$\tilde{\boldsymbol{z}}_{k+1}^{av} = \left(I - \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R\right)\tilde{\boldsymbol{z}}_k^{av} + \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*. \tag{6.34}$$

Conditions (ii) and (iii) follow from the above recursion and from the definition of $\alpha_c(\cdot)$. In fact,

$$f_{av}(\tilde{\boldsymbol{z}}_k^{av}) = -R\tilde{\boldsymbol{z}}_k^{av} - \alpha_c\big(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av}\big)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\tilde{\boldsymbol{z}}_k^{av} + \alpha_c\big(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av}\big)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*.$$

From the definition of $\alpha_c(\cdot)$, we obtain

$$\frac{\partial\alpha_c(a)}{\partial a} = \begin{cases} \frac{\alpha_0}{2}\left(\frac{a-C_1}{D}\right) & \text{if } C_1 < a < C_1 + D, \\ -\frac{\alpha_0}{2}\left(\frac{C_2-a}{D}\right) & \text{if } C_1 + D \le a < C_2, \\ 0 & \text{otherwise,} \end{cases}$$

and thus, the gradient of f_{av} is

$$\nabla_{\tilde{\boldsymbol{z}}} f_{av} = -R - \alpha_c\big(\boldsymbol{w}_{*,\,\bar{k}} - \tilde{z}^c_{k,\,\bar{k}}\big)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \frac{\partial \alpha_c}{\partial \tilde{z}^c_{k,\,\bar{k}}}\tilde{\boldsymbol{z}}^c_k \boldsymbol{e}_{\bar{k}}^T. \tag{6.35}$$

We now compute $\nabla_{\tilde{\boldsymbol{z}}} f$:

$$\nabla_{\tilde{\boldsymbol{z}}} f = -\boldsymbol{x}_k \boldsymbol{x}_k^T - \alpha_c\big(\boldsymbol{w}_{*,\,\bar{k}} - \tilde{z}^c_{k,\,\bar{k}}\big)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \frac{\partial \alpha_c}{\partial \tilde{z}^c_{k,\,\bar{k}}}\tilde{\boldsymbol{z}}^c_k \boldsymbol{e}_{\bar{k}}^T.$$

Condition (iii) follows from this relation and the fact that both $\alpha_c(\cdot)$ and its derivative are continuous and bounded functions.

We still need to check condition (i) before we can use Theorem 3.4, i.e., we need to prove that the origin $\tilde{\boldsymbol{z}}^{av}_k = \boldsymbol{0}$ is an exponentially stable equilibrium point of (6.30). Note first that $\boldsymbol{0}$ is an equilibrium point of (6.30) since, by definition, $\alpha_c(w_{*,\,\bar{k}}, 0) = 0$. To prove that it is exponentially stable, we shall proceed by showing that $\|\tilde{\boldsymbol{z}}^{av}_{k+1}\| \leq \gamma \|\tilde{\boldsymbol{z}}^{av}_k\|$ for all $k \geq 0$ and for some $\gamma < 1$.[6]

Before we evaluate the norm of $\tilde{\boldsymbol{z}}^{av}_{k+1}$, we need to relate the term $\mu\alpha_c(w_{*,\,\bar{k}} - \tilde{z}^{av}_{k,\,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*$ in (6.34) to $\tilde{\boldsymbol{z}}_k$. We do so by relating $\boldsymbol{w}_*$ to $\tilde{\boldsymbol{z}}_k$ as follows.

Let the constant $\eta_c > 0$ be such that C_1 satisfies

$$C_1 = (1 + \eta_c)\|\boldsymbol{w}_*\|_\infty. \tag{6.36}$$

Recall that the leakage term is nonzero if and only if $|w_{*,\,\bar{k}} - \tilde{z}^{av}_{k,\,\bar{k}}| > C_1$. Therefore, if $w_{*,\,\bar{k}} \cdot \tilde{z}^{av}_{k,\,\bar{k}} \geq 0$, $\alpha_c(w_{*,\,\bar{k}} - \tilde{z}^{av}_{k,\,\bar{k}}) \neq 0$ implies that

$$C_1 = (1 + \eta_c)\|\boldsymbol{w}_*\|_\infty < \big|w_{*,\,\bar{k}} - \tilde{z}^{av}_{k,\,\bar{k}}\big| = \big|\tilde{z}^{av}_{k,\,\bar{k}}\big| - \big|w_{*,\,\bar{k}}\big|,$$

and thus $\alpha_c \neq 0$ implies that $\big|\tilde{z}^{av}_{k,\,\bar{k}}\big| > (1 + \eta_c)\|\boldsymbol{w}_*\|_\infty$ (if $\tilde{z}^{av}_{k,\,\bar{k}}$ and $w_{*,\,\bar{k}}$ have the same sign). Repeating the argument for $w_{*,\,\bar{k}} \cdot \tilde{z}^{av}_{k,\,\bar{k}} < 0$, we conclude that

$$\alpha_c(w_{*,\,\bar{k}} - \tilde{z}^{av}_{k,\,\bar{k}}) \neq 0 \Rightarrow \begin{cases} |w_{*,\,\bar{k}}| < \frac{1}{1+\eta_c}|\tilde{z}^{av}_{k,\,\bar{k}}|, & \text{if } w_{*,\,\bar{k}} \cdot \tilde{z}^{av}_{k,\,\bar{k}} \geq 0, \\ |w_{*,\,\bar{k}}| < \frac{1}{\eta_c}|\bar{z}^c_{k,\,\bar{k}}|, & \text{if } w_{*,\,\bar{k}} \cdot \tilde{z}^{av}_{k,\,\bar{k}} < 0. \end{cases} \tag{6.37}$$

[6] This is a Lyapunov argument, where our choice for a Lyapunov function is $V(\tilde{\boldsymbol{z}}^{av}) = \|\tilde{\boldsymbol{z}}^{av}\|$.

This allows us to express $w_{*,\bar{k}}$ as $\epsilon_k \tilde{z}^{av}_{k,\bar{k}}$ for some ϵ_k in the interval $-\frac{1}{1+\eta_c} < \epsilon_k < \frac{1}{\eta_c}$. Using this result in (6.30), we obtain

$$\tilde{\boldsymbol{z}}^{av}_{k+1} = \left(I - \mu(1+\epsilon_k)\alpha_c(w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}^T_{\bar{k}} - \mu R\right)\tilde{\boldsymbol{z}}^{av}_k \ .$$

Introduce the coefficient matrix $A(\epsilon_k) = I - \mu(1+\epsilon_k)\alpha_c(\cdot)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}^T_{\bar{k}} - \mu R$. We now show that $A(\epsilon_k)$ is uniformly contractive, i.e., $\|A(\epsilon_k)\| \leq \gamma < 1$ for all k (note that, since A is symmetric, $\|A\| = \left|\lambda_{\max}(A)\right|$ and we can thus show instead that the eigenvalues of $A(\epsilon_k)$ are uniformly upper bounded by 1).

So let $\boldsymbol{y}$ be a vector with unit norm, and compute

$$\boldsymbol{y}^T A(\epsilon_k)\boldsymbol{y} = \boldsymbol{y}^T(I - \mu R)\boldsymbol{y} - \mu(1+\epsilon_k)\alpha_c(\cdot)(\boldsymbol{y}^T\boldsymbol{e}_{\bar{k}})^2.$$

If $\alpha_c = 0$, $A(\epsilon_k) = I - \mu R$, and $|\lambda(A)| < 1$ if and only if

$$\mu\lambda_{\min}(R) > 0, \qquad \text{and} \qquad \mu\lambda_{\max}(R) < 2,$$

which are the usual conditions for the mean-square stability of LMS. On the other hand, if $\alpha_c = \rho\alpha_0$, for some $0 \leq \rho \leq 1$, we have

$$1 - \mu\lambda_{\max}(R) - \mu\rho\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < \boldsymbol{y}^T A(\epsilon_k)\boldsymbol{y} < \\ < 1 - \mu\lambda_{\min}(R) - \mu\rho\left(1 - \frac{1}{1+\eta_c}\right)\alpha_0 \ .$$

Therefore, $|\lambda[A(\epsilon_k)]| < 1$ if

$$\mu\lambda_{\min}(R) + \mu\left(1 - \frac{1}{1+\eta_c}\right)\alpha_0 > 0,$$

and

$$\mu\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < 2 - \mu\lambda_{\max}(R). \tag{6.38}$$

The first of these conditions is always satisfied since, by assumption, $\eta_c > 0$ and $\lambda_{\min}(R) > 0$. The second condition provides an upper bound on α_0, as a function of our choices for μ and η_c (or C_1). In this case, we obtain

$$\|A(\epsilon_k)\| < \max\left\{1 - \mu\left(1 - \frac{1}{1+\eta_c}\right)\alpha_0 - \mu\lambda_{\min}(R), \right.$$
$$\left. \mu\left(1 + \frac{1}{\eta_c}\right)\alpha_0 + \mu\lambda_{\max}(R) - 1\right\} \triangleq \gamma < 1 \ .$$

It then follows that

$$\|\bar{\boldsymbol{z}}_{k+1}^c\| < \|A(\epsilon_k)\| \ \|\bar{\boldsymbol{z}}_k^c\| \leq \gamma\|\bar{\boldsymbol{z}}_k^c\|,$$

and thus $\tilde{\boldsymbol{z}}_k^{av} = \mathbf{0}$ is an exponentially stable equilibrium point of (6.30).

With this result, we can apply Theorem 3.4 and conclude that the steady-state of circular-leaky can be obtained from the partially averaged recursion

$$\tilde{\boldsymbol{z}}_{k+1}^{pav} = (I - \mu R)\tilde{\boldsymbol{z}}_k^{pav} - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\delta}_k^c. \tag{6.39}$$

The covariance of $\tilde{\boldsymbol{z}}_k^{pav}$ is

$$\bar{Z}_{k+1}^{pav} \triangleq \mathrm{E}\,\tilde{\boldsymbol{z}}_{k+1}^{pav}\tilde{\boldsymbol{z}}_{k+1}^{pav\,T} = (I - \mu R)\bar{Z}_k^{pav}(I - \mu R) + \mu^2\sigma_v^2 R + \sigma_d^2(I + R + \mu M\sigma_d^2 R).$$

Solving this recursion for small μ, we obtain

$$\bar{Z}_\infty^{pav} = \frac{\mu}{2}(\sigma_v^2 + M\sigma_d^2)I + \frac{1}{2\mu}\sigma_d^2\left(I + R^{-1}\right).$$

Theorem 3.4 says that the true covariance matrix, $\bar{Z}^c \triangleq \mathrm{E}\,\tilde{\boldsymbol{z}}_k^c\tilde{\boldsymbol{z}}_k^{c\,T}$, converges to the same steady-state as $\bar{Z}_k^{pav}$ if the step-size is sufficiently small.

For the MSE, we use the fact from Thm. 3.4 that

$$\sup_{k\geq 0} \mathrm{E}\,\|\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav}\|^2 \to 0, \tag{6.40}$$

as $\mu \to 0$. The computation of the MSE will be performed in several steps: first, note that

$$e(k) = \boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c + v(k),$$

and from the independence of $\{v(k)\}$ and $\{\boldsymbol{x}_k\}$, we obtain

$$\mathrm{E}\, e(k)^2 = \mathrm{E}\,(\boldsymbol{x}_k \tilde{\boldsymbol{z}}_k^c)^2 + \sigma_v^2. \tag{6.41}$$

To compute $\mathrm{E}\,(\boldsymbol{x}_k \tilde{\boldsymbol{z}}_k^c)^2$, we show that (in steady-state) this expectation is equal to $\mathrm{E}\,(\boldsymbol{x}_k \tilde{\boldsymbol{z}}_k)^2$, where $\tilde{\boldsymbol{z}}_k$ is the weight error obtained from the LMS algorithm with the same input and noise sequences.

This is shown as follows: notice that the above recursion for $\tilde{\boldsymbol{z}}_k^{pav}$, obtained for circular-leaky, is the same partially-averaged recursion that would be obtained for LMS. Therefore, Theorem 3.4 also implies that

$$\sup_{k \geq 0} \mathrm{E}\, \|\tilde{\boldsymbol{z}}_k - \tilde{\boldsymbol{z}}_k^{pav}\|^2 \to 0,$$

as $\mu \to 0$, where now $\tilde{\boldsymbol{z}}_k$ is the weight error computed by LMS. From this relation and (6.40), we conclude that

$$\sup_{k \geq 0} \mathrm{E}\, \|\tilde{\boldsymbol{z}}_k - \tilde{\boldsymbol{z}}_k^{c}\|^2 \to 0,$$

as $\mu \to 0$.

Next, note that

$$\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c\right)^2 - \left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^{pav}\right)^2 = \boldsymbol{x}_k^T \left(\tilde{\boldsymbol{z}}_k^c \tilde{\boldsymbol{z}}_k^{c\,T} - \tilde{\boldsymbol{z}}_k^{pav} \tilde{\boldsymbol{z}}_k^{pav\,T}\right) \boldsymbol{x}_k \leq$$

$$\leq \left\| \tilde{\boldsymbol{z}}_k^c \tilde{\boldsymbol{z}}_k^{c\,T} - \tilde{\boldsymbol{z}}_k^{pav} \tilde{\boldsymbol{z}}_k^{pav\,T} \right\| \|\boldsymbol{x}_k\|^2 \leq \left\| \tilde{\boldsymbol{z}}_k^c \tilde{\boldsymbol{z}}_k^{c\,T} - \tilde{\boldsymbol{z}}_k^{pav} \tilde{\boldsymbol{z}}_k^{pav\,T} \right\| B_x.$$

In addition,

$$\left\| \tilde{\boldsymbol{z}}_k^c \tilde{\boldsymbol{z}}_k^{c\,T} - \tilde{\boldsymbol{z}}_k^{pav} \tilde{\boldsymbol{z}}_k^{pav\,T} \right\| =$$

$$= \left\| (\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav})\,(\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav})^T - \tilde{\boldsymbol{z}}_k^{pav}\,(\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav})^T - (\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav})\,\tilde{\boldsymbol{z}}_k^{pav\,T} \right\| \leq$$

$$\leq \|\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav}\|^2 + 2\|\tilde{\boldsymbol{z}}_k^{pav}\|\,\|\tilde{\boldsymbol{z}}_k^c - \tilde{\boldsymbol{z}}_k^{pav}\|.$$

Since $\boldsymbol{x}_k$ and $v(k)$ are bounded, $\|\tilde{\boldsymbol{z}}_k^{pav}\|$ is also a bounded sequence for small μ. This fact and the general inequality for random variables $(\mathrm{E}\, a)^2 \leq \mathrm{E}\, a^2$, imply that

$$\lim_{k\to\infty} \sup_{k\geq 0} \mathrm{E}\left[\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c\right)^2 - \left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^{pav}\right)^2\right] \leq 0,$$

as $\mu \to 0$. Similar arguments show that

$$\lim_{k\to\infty} \sup_{k\geq 0} -\mathrm{E}\left[\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c\right)^2 - \left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^{pav}\right)^2\right] \leq 0,$$

We conclude that, for small μ and in steady-state,

$$\mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c\right)^2 \approx \mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^{pav}\right)^2.$$

We can repeat this argument, replacing $\tilde{\boldsymbol{z}}_k^c$ with $\tilde{\boldsymbol{z}}_k$, to conclude that, in steady-state,

$$\mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^c\right)^2 \approx \mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k^{pav}\right)^2 \approx \mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k\right)^2. \tag{6.42}$$

The quantity on the right-hand side is the value obtained from the LMS algorithm. We can use an argument from the proof of Thm. 3.5 in [Sol89] to obtain the MSE formula in (6.33), as follows.

Let the covariance of $\tilde{\boldsymbol{z}}_k$ be $\bar{Z}_k \stackrel{\Delta}{=} \mathrm{E}\, \tilde{\boldsymbol{z}}_k \tilde{\boldsymbol{z}}_k^T$. We know from the above arguments that this covariance reaches a steady-state, so we can write

$$\begin{aligned}\mathrm{Tr}\left(\bar{Z}_\infty\right) = \lim_{k\to\infty} \Big\{ &\mathrm{Tr}\left(\mathrm{E}\left[\left(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\right) \tilde{\boldsymbol{z}}_k \tilde{\boldsymbol{z}}_k^T \left(I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\right)\right]\right) + \\ &+ \mu^2 \sigma_v^2\, \mathrm{E}\,\|\boldsymbol{x}_k\|^2 + \mathrm{Tr}\left(\Sigma_\xi + \mu^2 \sigma_d^2 M R\right)\Big\},\end{aligned}$$

where we already ignored the cross-terms. Expanding the first term in the right-hand side, we obtain

$$\lim_{k\to\infty} \mathrm{E}\left(\boldsymbol{x}_k^T \tilde{\boldsymbol{z}}_k\right)^2 = \frac{1}{2}\left\{\mu\left(\sigma_v^2 + M\sigma_d^2\right)\mathrm{Tr}(R) + \frac{1}{2\mu}\mathrm{Tr}\left(\Sigma_\xi\right)\right\}$$

Combining this result with (6.41) and (6.42), we obtain the MSE formula (6.33) in the statement of the theorem.

$\diamondsuit$

This theorem shows that circular-leaky has essentially the same good performance as LMS if $R > 0$ and condition (6.31) is satisfied. Therefore, the parameters α_0, μ, and η_c must be chosen so that (6.31) holds. We provide design examples in Secs. 6.8 and 6.9.

6.6.2 The Modified Switching-σ Algorithm

A similar result is developed here for the modified switching-σ algorithm. The conditions for Theorem 6.3 below are less restrictive than for circular-leaky (compare the condition for circular-leaky, (6.31), with the condition $S_1 > \|\boldsymbol{w}_*\|$ for switching-σ). [The finite-precision error equation for switching-σ is given in Appendix 6.D.]

Theorem 6.3 (Performance of switching$-\sigma$). *Assume that $\{\boldsymbol{x}_k\}$, $\{v(k)\}$, and $\{\boldsymbol{\delta}_k^c\}$ are stationary, have zero mean, and satisfy $\mathrm{E}\,\boldsymbol{x}_k\boldsymbol{x}_k^T = R > 0$. Assume further that $\{v(k)\}$ is iid and independent of $\{\boldsymbol{x}_k\}$, and that Assumption UM-1 holds. Then, if the step-size μ is small enough and $S_1 > \|\boldsymbol{w}_*\|$, the switching-$\sigma$ estimates $\boldsymbol{z}_k^s$ are asymptotically unbiased, and in steady-state we have*

$$\lim_{k\to\infty} \mathrm{E}\,\|\tilde{\boldsymbol{z}}_k\|^2 \approx \frac{\mu}{2}\left(\sigma_v^2 + \sigma_d^2\right) M + \frac{\sigma_d^2}{2\mu}\,\mathrm{Tr}\left(I + R^{-1}\right).$$

$$\lim_{k\to\infty} \mathrm{E}\,e^s(k)^2 \approx \sigma_v^2 + \mu(\sigma_v^2 + \sigma_d^2)\frac{\mathrm{Tr}(R)}{2} + \frac{\mathrm{Tr}(\sigma_d^2(I+R))}{2}\,. \tag{6.43}$$

Proof: As we did for Theorem 6.2, we need to check conditions (i)–(iii) from Theorem 3.4. Conditions (ii) and (iii) can be checked as before, but a stronger

result can be obtained if we modify the argument for checking condition (i). Indeed, instead of working with the averaged error $\tilde{\boldsymbol{z}}_k^{av}$, we now work with the averaged version of $\boldsymbol{z}_k^s$, namely

$$\boldsymbol{z}_{k+1}^{av} = \big((1-\alpha_s(\boldsymbol{z}_k^{av}))I - \mu R\big)\boldsymbol{z}_k^{av} + \mu R\boldsymbol{w}_*. \tag{6.44}$$

Condition (i) is equivalent to proving that $\boldsymbol{w}_*$ is an exponentially-stable equilibrium point for (6.44).

We show in Appendix 6.E that there exists a K such that $\|\boldsymbol{z}_k^{av}\| < S_1$ for all $k \geq K$. Therefore, for large k, the leakage term remains equal to zero ($\alpha_s(\boldsymbol{z}_k^{av}) = 0$) and the averaged recursion (6.44) becomes

$$\boldsymbol{z}_{k+1}^{av} = \big(I - \mu R\big)\boldsymbol{z}_k^{av} + \mu R\boldsymbol{w}_*,$$

from which we conclude that $\boldsymbol{z}_k^{av} \to \boldsymbol{w}_*$ exponentially fast if μ satisfies $0 < \mu\lambda(R) < 2$. Having verified that condition (i) is satisfied, we can then apply Theorem 3.4 to obtain (6.43), just as we did for circular-leaky in Theorem 6.2.

$\diamondsuit$

6.7 DETERMINISTIC STABILITY ANALYSIS

Having shown that the circular-leaky and switching-σ algorithms do not introduce bias, we now prove that they also avoid drift for any bounded input and noise sequences (provided that the step-size is small enough). We analyze switching-σ first (the stability proof for circular-leaky is considerably more involved).

Theorem 6.4 (Stability of switching-σ). *If μ satisfies*

$$|1 - \mu\alpha_0 - \mu B_x| \leq |1 - \mu\alpha_0| < 1, \tag{6.45}$$

then the fixed-point switching-σ algorithm is bounded-input bounded-state stable (with the $v(k)$ as input and $\boldsymbol{z}_k^s$ as the state).

Proof: From Appendix 6.D, the fixed-point recursion for the switching-σ algorithm is

$$\boldsymbol{z}_{k+1}^s = \Big((1-\mu\alpha_s(\cdot))I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\Big)\boldsymbol{z}_k^s + \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\boldsymbol{w}_* + \mu\boldsymbol{x}_k v(k) + \boldsymbol{\delta}_k^s. \tag{6.46}$$

Our goal is to show that the sequence $\big\{\|\boldsymbol{z}_k^s\|\big\}_{k=0}^{\infty}$ is bounded. The first task in the proof is to find a ball $\mathcal{B}$ (centered at the origin), outside of which the norm $\|\boldsymbol{z}_k^s\|$ is strictly decreasing, i.e.,

$$\|\boldsymbol{z}_{k+1}^s\| - \|\boldsymbol{z}_k^s\| < 0 \qquad \text{if } \boldsymbol{z}_k^s \notin \mathcal{B}.$$

We do not need to find the smallest ball satisfying the above property to prove that $\big\{\|\boldsymbol{z}_k^s\|\big\}_{k=0}^{\infty}$ is bounded, we only need to find one such ball. With this in mind, our argument is simplified if we restrict ourselves to balls $\mathcal{B}_r$ with radii $r > S_2$, so that $\alpha_s(\boldsymbol{z}) = \alpha_0$ for any $\boldsymbol{z} \notin \mathcal{B}_r$.

Assume then that $\boldsymbol{z}_k^s$ satisfies $\|\boldsymbol{z}_k^s\| > S_2$ at some instant k. Taking norms of both sides of (6.46), and using (6.45) to upper bound $\big\|\big[(1-\mu\alpha_0)I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\big]\big\|$, we obtain the inequality

$$\begin{aligned}\|\boldsymbol{z}_{k+1}^s\| \le |1-\mu\alpha_0|\|\boldsymbol{z}_k^s\| + \mu B_x\|\boldsymbol{w}_*\| + \\ + \mu\sqrt{B_x}v_{\max} + \Big((2+\mu\alpha_0) + \mu\sqrt{B_x}\Big)\sqrt{M}\varepsilon,\end{aligned} \tag{6.47}$$

where we also used (6.3) to bound $v(k)$ and $\boldsymbol{x}_k$, and the bound for $\|\boldsymbol{\delta}_k^s\|$ from Appendix 6.D. Subtracting $\|\boldsymbol{z}_k^s\|$ from both sides of (6.47), we obtain

$$\|\boldsymbol{z}_{k+1}^s\| - \|\boldsymbol{z}_k^s\| \le -\mu\alpha_0\|\boldsymbol{z}_k^s\| + \mu B_x\|\boldsymbol{w}_*\| + \mu\sqrt{B_x}v_{\max} + \Big((2+\mu\alpha_0) + \mu\sqrt{B_x}\Big)\sqrt{M}\varepsilon.$$

From this inequality, and from our assumption that $\|\boldsymbol{z}_k^s\| > S_2$, it follows that $\|\boldsymbol{z}_{k+1}^s\| < \|\boldsymbol{z}_k^s\|$ if

$$\|\boldsymbol{z}_k^s\| > \Omega \triangleq \max\left\{S_2, \quad \frac{\mu B_x\|\boldsymbol{w}_*\| + \mu\sqrt{B_x}v_{\max} + \Big((2+\mu\alpha_0) + \mu\sqrt{B_x}\Big)\sqrt{M}\varepsilon}{\mu\alpha_0}\right\}. \tag{6.48}$$

We therefore can choose $\mathcal{B} = \{\boldsymbol{z} : \ \|\boldsymbol{z}\| \leq \Omega\}$. To complete our argument, note the following:

1. We may have $\|\boldsymbol{z}_{n+1}^s\| \geq \|\boldsymbol{z}_n^s\|$ only if $\boldsymbol{z}_n^s \in \mathcal{B}$. However, $\|\boldsymbol{z}_{n+1}^s\|$ cannot be arbitrarily large. In fact, using the switching-σ recursion (6.46) we can evaluate the worst-case $\|\boldsymbol{z}_{n+1}^s\|$ (the bound below is not tight):

$$\sup_{\boldsymbol{z}_n^s \in \mathcal{B}} \|\boldsymbol{z}_{n+1}^s\| \leq \Omega + \mu B_x \|\boldsymbol{w}_*\| + \mu\sqrt{B_x}\, v_{\max} + \left((2+\mu\alpha_0) + \mu\sqrt{B_x}\right)\sqrt{M}\varepsilon. \tag{6.49}$$

2. If $\boldsymbol{z}_n^s$ is not inside $\mathcal{B}$ at a particular time instant n (i.e., $\|\boldsymbol{z}_n^s\| > \Omega$), then $\|\boldsymbol{z}_{n+1}^s\| < \|\boldsymbol{z}_n^s\|$. Repeating this argument, we conclude that either $\|\boldsymbol{z}_k^s\| < \|\boldsymbol{z}_n^s\|$ for all $k > n$, or there exists a time (say, N) such that $\boldsymbol{z}_{n+N}^s \in \mathcal{B}$.

The result of the theorem follows from these two observations.

$\diamondsuit$

The stability analysis of the circular-leaky algorithm is similar in spirit to that for switching$-\sigma$. However, the fact that leakage is applied (or not) to only one tap at each time instant in a pre-specified circular order requires a closer study to prove stability. This is because it can happen that $\|\boldsymbol{z}_k^c\|_\infty$ is large, but the tap that is being checked by $\alpha_c(\cdot)$ at time k (i.e., $z_{k,\bar{k}}^c$) is small, so that no leakage is applied. One then needs to verify that such possibilities do not cause instability. To account for this scenario, we need to look at the variation of the norm of $\boldsymbol{z}_k^c$ after M time-steps, i.e., we compare $\|\boldsymbol{z}_{k+M}^c\|$ with $\|\boldsymbol{z}_k^c\|$. The reassuring conclusion is that circular-leaky is also stable — see Appendix 6.F.

Theorem 6.5 (Stability of circular leaky). *If μ satisfies*

$$|1-\mu\alpha_0-\mu B_x| < 1 \qquad \text{and} \qquad \sqrt{1-\frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + \mu M B_x < 1\,, \tag{6.50}$$

then the fixed-point circular-leaky algorithm is bounded-input bounded-state stable.

$\diamondsuit$

6.8 SIMULATION RESULTS

We now present several simulation results. We first apply the circular-leaky algorithm to the example of drift shown in Fig. 6.1. In that example, we had $\mu = 0.15$, $\|\boldsymbol{w}_*\|_\infty = 0.44$, and $\mu\lambda_{max}(R) = 0.075$. As in Sec. 6.1.1, we implemented the algorithms in fixed-point with 7 bits plus sign.

To choose the parameters for the circular-leaky algorithm, we need a bound on $\|\boldsymbol{w}_*\|_\infty$. Assume that the bound $\|\boldsymbol{w}_*\|_\infty \leq 0.55$ is given. Choosing $\mu\alpha_0 = 0.1$, (6.31) requires that $\eta_c > 0.055$ and thus we need $C_1 > 0.58$. We chose $C_1 = 0.60$ and $C_2 = 0.61$. The results are shown in Fig. 6.4, where we plotted $\|\boldsymbol{z}_k\|_\infty$ for circular-leaky, LMS, and for leaky LMS with $\mu\alpha_0 = 0.0156$ (note that for fixed-point numbers with 7 bits plus sign, this value of $\mu\alpha_0$ is only the second smallest representable number). Since in this example the input distribution does not satisfy $R > 0$, the LMS algorithm overflows, as we saw in Sec. 6.1.1. Circular-leaky prevents the overflow, keeping the estimates at a safe level. The squared error curves $e^l(k)^2$ and $e^c(k)^2$ are presented in Fig. 6.5, where we see that the error level is significantly smaller for circular-leaky than for leaky LMS.

In Fig. 6.6 we plot the ensemble-averaged learning curves computed by the same algorithms, when $R = \text{diag}(0.25,\ 0.25)$ (since the step-size is small in this example, we averaged the curves over 100 experiments). Note that the performance of the leaky LMS algorithm is considerably worse, even though we have used the second smallest value for $\mu\alpha_0$.

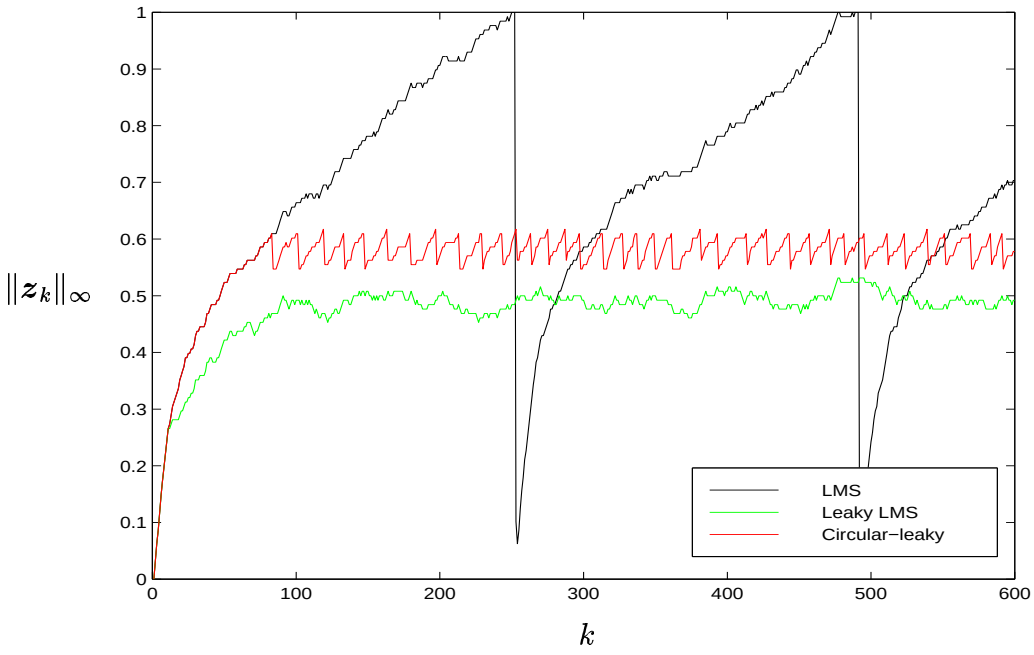

Figure 6.4: *Application of the LMS, leaky LMS and circular-leaky algorithms to the example of Fig. 6.1.*

We now present two examples to highlight the robustness of circular-leaky. In the first one we used $M = 10$ and again $\mu = 0.15$, $\mu\alpha_0 = 0.1$, $C_1 = 0.60$, $C_2 = 0.61$, and $\|\boldsymbol{w}_*\|_\infty = 0.44$. The input sequence has covariance matrix with 9 zero eigenvalues, and one eigenvalue equal to 2.5. We also artificially added 1 LSB (i.e., one least-significant bit, in this case equal to $1/128$) to every entry of $\boldsymbol{z}_k$ at every time step, in order to make the task of circular-leaky and leaky LMS more challenging. In Fig. 6.7 we plotted $\|\boldsymbol{z}_k\|_\infty$ for LMS and circular-leaky. The discontinuities in the LMS plot correspond to points where overflow occurs; circular-leaky avoids overflow even in this demanding environment.

The last example has $M = 100$, and was implemented with 11 bits plus sign. The input and true weight were

$$\boldsymbol{w}_* = \begin{bmatrix} 0.06 & -0.06 & 0.06 \ldots & -0.06 \end{bmatrix}^T$$

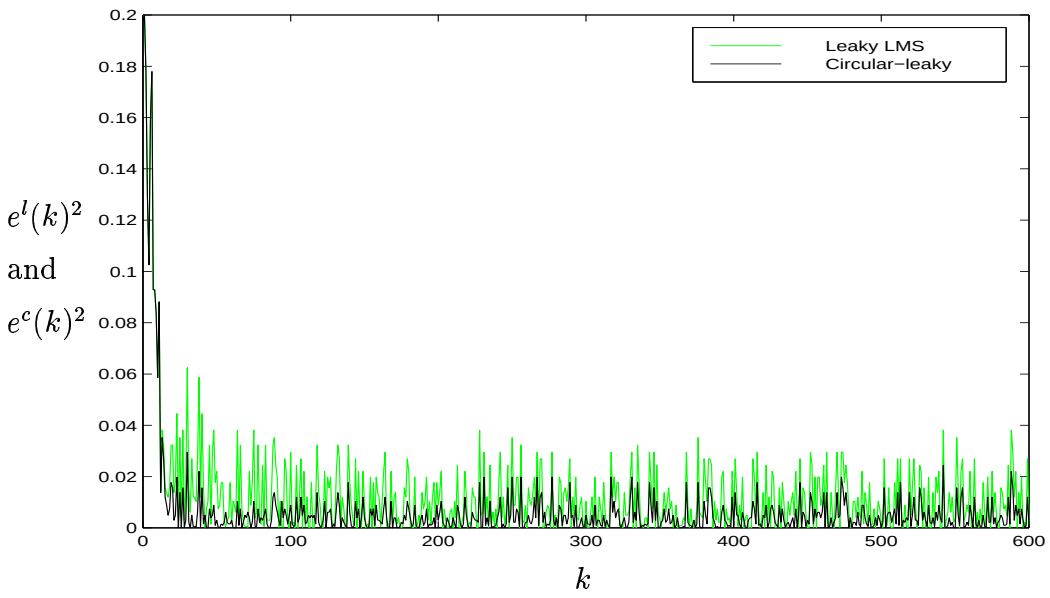

Figure 6.5: *Squared error curves for leaky LMS and circular-leaky in the same example as in Fig. 6.1.*

and

$$\boldsymbol{x}_k = \begin{bmatrix} \pm 0.5 & \mp 0.5 & \pm 0.5 & \cdots \mp 0.5 \end{bmatrix}^T$$

The input correlation matrix had 99 zero eigenvalues and one eigenvalue equal to 25. The other parameters were $\mu = 0.01$, $\mu\alpha_0 = 0.1$, $\sigma_v^2 = 1/3 \times 10^{-3}$, $C_1 = 0.21$, and $C_2 = 0.22$. The plots of $\|\boldsymbol{z}_k\|_\infty$ (LMS) and $\|\boldsymbol{z}_k^c\|_\infty$ (circular-leaky) are shown in Fig. 6.8.

6.9 FILTER DESIGN

In order to choose the design parameters for the circular-leaky algorithm (6.22), a bound $W_\infty \geq \|\boldsymbol{w}_*\|_\infty$ is necessary. This norm could be obtained from approximations for the statistics of the signals involved. For example, if we know that

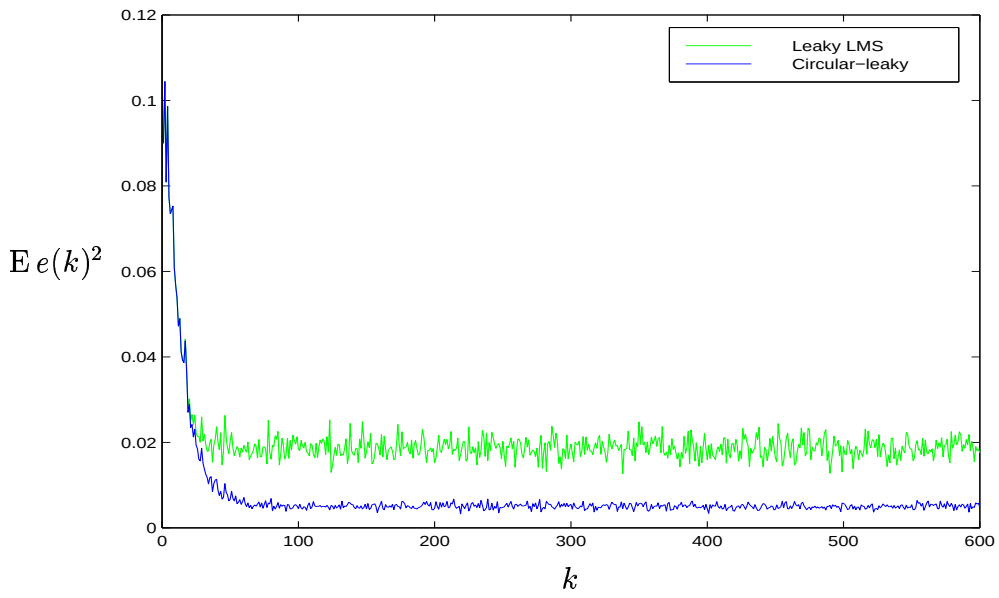

Figure 6.6: *Learning curves ($e(k)^2$ averaged over 100 runs) for LMS, leaky LMS and circular-leaky, with* $R = \mathrm{diag}(0.25, 0.25)$.

the true covariance $R = \mathrm{E}\,\boldsymbol{x}_k \boldsymbol{x}_k^T$ and cross-correlation $\boldsymbol{p} = \mathrm{E}\,y(k)\boldsymbol{x}_k$ are inside balls

$$R = \tilde{R} + \delta R, \qquad\qquad \boldsymbol{p} = \tilde{\boldsymbol{p}} + \boldsymbol{\delta p},$$

with $\|\delta R\|_\infty \leq \eta_R$, $\|\boldsymbol{\delta p}\|_\infty \leq \eta_p$, a bound for $\|\boldsymbol{w}_*\|_\infty$ could be computed from

$$\|\boldsymbol{w}_*\|_\infty = \left\|R^{-1}\boldsymbol{p}\right\|_\infty = \left\|\left(\tilde{R} + \delta R\right)^{-1}(\tilde{\boldsymbol{p}} + \boldsymbol{\delta p})\right\|_\infty .$$

The matrix inversion lemma of Appendix 2.A.2 can be applied to obtain

$$\|\boldsymbol{w}_*\|_\infty \leq \left\|\tilde{R}^{-1}\tilde{\boldsymbol{p}} - \tilde{R}^{-1}(I + \delta R)^{-1}\delta R\tilde{R}^{-1}\tilde{\boldsymbol{p}} + \tilde{R}^{-1}\boldsymbol{\delta p}\right\| + O(\eta_R\eta_p).$$

Assuming that $\eta_R \ll 1$ so that $(I + \delta R)^{-1} \approx I$, we obtain

$$\|\boldsymbol{w}_*\|_\infty \leq \left\|\tilde{R}^{-1}\tilde{\boldsymbol{p}}\right\|_\infty + \left\|\tilde{R}^{-2}\tilde{\boldsymbol{p}}\right\|_\infty \eta_R + \left\|\tilde{R}^{-1}\right\|_\infty \eta_p.$$

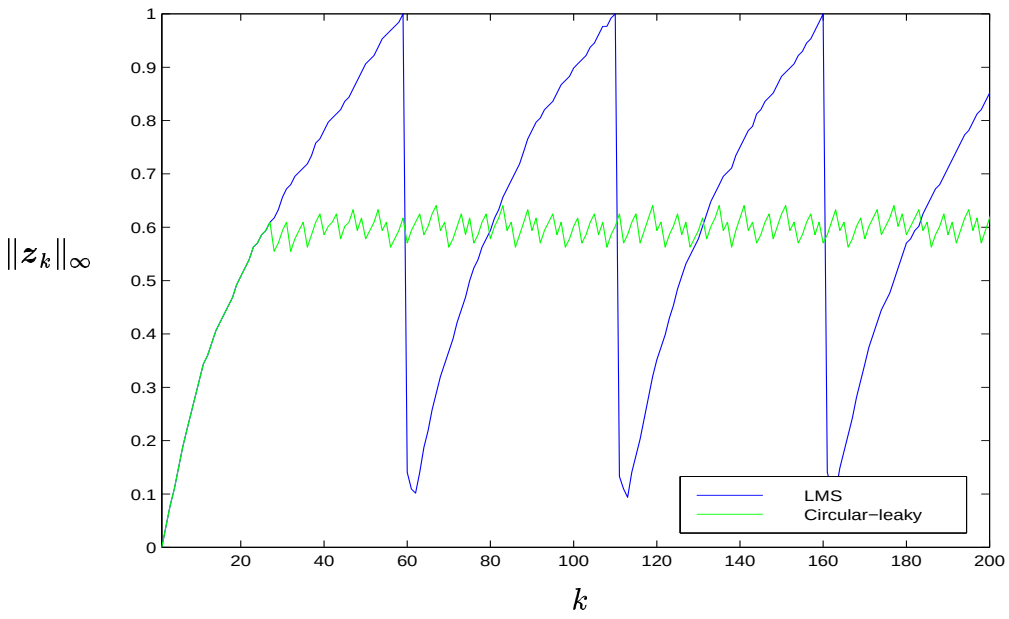

Figure 6.7: $\|\boldsymbol{z}_k\|_\infty$ *for LMS and circular-leaky, with* $M = 10$, $\mu\alpha_0 = 0.1$ *and* $C_1 = 0.60$, $C_2 = 0.61$.

Assuming that a bound for $\|\boldsymbol{w}_*\|_\infty$ is available, the design is made in the following way. Begin by choosing adequate values for μ and α_0 satisfying $0 < \mu\alpha_0 < 1$, and use (6.31) to find the smallest possible η_c. The parameter C_1 is then chosen from $C_1 \geq (1 + \eta_c)W_\infty$. If the resulting C_1 is too large, we can reduce α_0 or μ or both to allow for a smaller η_c in (6.31), and repeat the above procedure.

This procedure guarantees that circular-leaky is unbiased if the step-size used is small enough (unfortunately, as always with the use of averaging results, we cannot tell how small must μ be).

Although it is not a necessary condition, (6.31) is not excessively conservative. We also proved that circular-leaky is stable if condition (6.50) is satisfied. This condition however is conservative, the filter may be stable even if the condition is not satisfied.

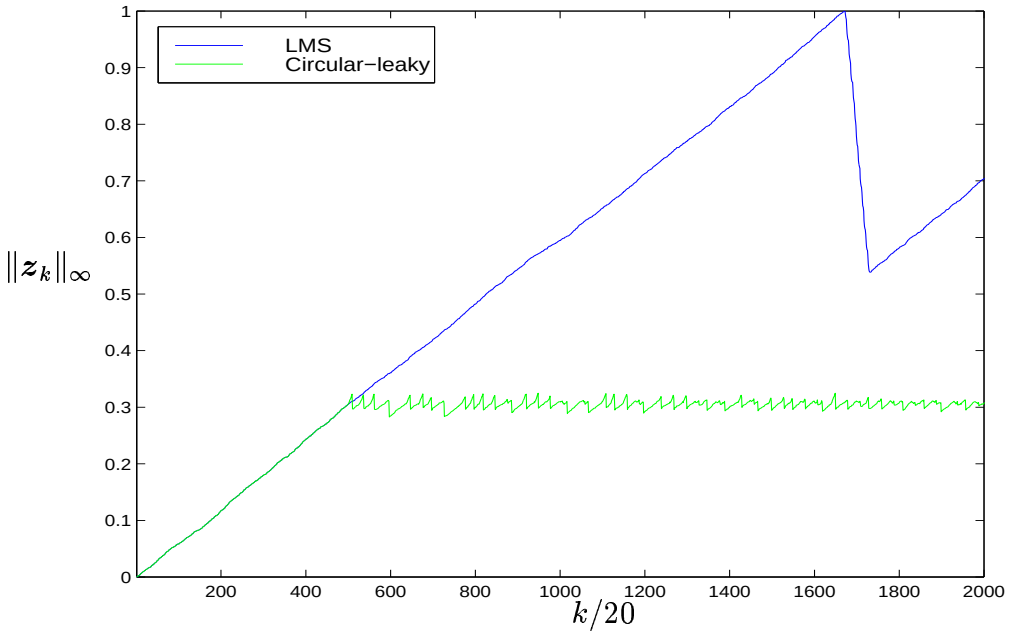

Figure 6.8: $\|\boldsymbol{z}_k\|_\infty$ *for LMS and circular-leaky, with* $M = 100$, $\mu\alpha_0 = 0.1$. *Only one out of every* 20 *samples is plotted.*

6.10 CONTRIBUTIONS OF THIS CHAPTER

The main result in this chapter is the introduction of a new variant of LMS that prevents the occurrence of the drift phenomenon, but avoids the higher computational cost and the bias introduced by leaky LMS (which is the standard solution to drift). This chapter follows the presentation in [NS99c].

A standing open issue in the adaptive filtering literature (as seen in the quote from [Set93] in Sec. 1.4), has been the need for a precise analysis of the leaky LMS algorithm (i.e., an analysis that does not require the independence assumptions). Although we did not address this problem here, we argued that it becomes virtually meaningless since our new algorithm outperforms leaky LMS, and our analysis does not require independence of the input sequence $\{\boldsymbol{x}_k\}$.

A new stochastic analysis, and a new deterministic analysis for fixed-point implementations of a modified version of the switching-σ algorithm were also provided in this chapter. Switching-σ is an algorithm used in the adaptive control literature [IK84, IT86, IS96] to avoid drift without the introduction of bias, although this last fact was only proved for deterministic square-sumable (ℓ_2) noise sequences before. Finite-precision implementations of the algorithm were also not available until the results presented here and in [NS96, NS99c].

In addition, we also extended the deterministic analysis of LMS, providing a stability analysis for floating-point implementations.

There is a vast literature for the deterministic analysis of the LMS algorithm, mainly for infinite-precision arithmetic. Most results are concerned with the drift phenomenon, and study in detail when and how it can occur [IK84, SLJ86, IT86, WS90, RBJ91, SK95, Rup95]. A deterministic analysis of finite-precision implementations of LMS appeared in [WM79]. This paper assumes that the computations are done in a combination of fixed and floating-point, but some important features of the floating-point model, as the error proportional to $\|\boldsymbol{z}_k\|$ in (6.12), are missing. Moreover, [WM79] does not provide bounds for the errors based on the machine precision, as we do in Theorem 6.1.

A point that deserves further investigation is the choice of the leakage function $\alpha_c(\cdot)$. Our choice of a differentiable $\alpha_c(\cdot)$ was motivated by the fact that the averaging results of Theorem 3.4 are not applicable to discontinuous $f(k, \tilde{\boldsymbol{w}}_k)$. The stability results of Sec. 6.7, however, are still valid if instead of (6.23), we choose a hard-limiting α_c, say one of the form

$$\alpha_c(a) = \begin{cases} \alpha_0 & \text{if } |a| > C_1, \\ 0 & \text{if } |a| \leq C_1. \end{cases}$$

This is of course a much simpler function to implement than (6.23) — see [NS96].

The stability theorem (Thm. 6.5) needs almost no modification to allow for this different definition of α_c. Unfortunately, the situation is not as simple for the stochastic averaging analysis. It is possible to perform an entirely deterministic analysis and show that, if the input sequence $\{\boldsymbol{x}_k\}$ is sufficiently PE and the noise $v(k)$ and machine precision ε are sufficiently small, then $\alpha_c \equiv 0$ for large k, and the circular-leaky algorithm with the above function $\alpha_c(\cdot)$ will perform as the LMS algorithm [NS96]. Nevertheless, the conditions for the machine precision and step-size that arise from this analysis are not easily computable (and must be recomputed for each different input sequence). Therefore, they do not provide design conditions for the choice of the filter parameters (α_0, C_1, and μ).

APPENDICES FOR CHAPTER 6

6.A YOUNG'S INEQUALITY

The following simple version of Young's inequality [Mit70] is used in Sec. 6.4. For any $a,\ b \in \mathbb{R}$ and $\rho > 0$,

$$a \cdot b \ \leq \ \rho\, a^2 + \frac{1}{\rho}\, b^2. \tag{6.A.1}$$

6.B PROOF OF LEMMA 6.1

First note that from $\mu B_x < 2$, and using the fact that $a \leq \mu N B_x$, it can be shown that $N\gamma < 1$. The following proof derives an expression for the rate of convergence of LMS. In particular, (6.B.3) is a new result.

Define $\Phi(i,j)$ to be the state transition matrix of the linear system (6.5):

$$\Phi(i,j) = \prod_{n=j}^{i-1} \Big(I - \mu(n) \boldsymbol{x}_n \boldsymbol{x}_n^T \Big), \quad \Phi(j,j) = I. \tag{6.B.1}$$

Then $\tilde{\boldsymbol{w}}_i = \Phi(i,k)\tilde{\boldsymbol{w}}_k$ for $i \geq k$. It can be verified that

$$\|\tilde{\boldsymbol{w}}_{i+1}\|_2^2 - \|\tilde{\boldsymbol{w}}_i\|_2^2 = -\mu(i)\big(\tilde{\boldsymbol{w}}_i^T \boldsymbol{x}_i\big)^2 \big(2 - \mu(i)\boldsymbol{x}_i^T \boldsymbol{x}_i\big) \leq -\mu(i)\big(\tilde{\boldsymbol{w}}_i^T \boldsymbol{x}_i\big)^2 (2 - \mu B_x).$$

Add this inequality over N steps,

$$\begin{aligned} \|\tilde{\boldsymbol{w}}_{k+N}\|^2 - \|\tilde{\boldsymbol{w}}_k\|^2 &\leq -(2 - \mu B_x) \sum_{i=k}^{k+N-1} \mu(i)\big(\tilde{\boldsymbol{w}}_i^T \boldsymbol{x}_i\big)^2 \\ &= -(2 - \mu B_x)\tilde{\boldsymbol{w}}_k^T \left[\sum_{i=k}^{k+N-1} \Phi(i,k)^T \boldsymbol{x}_i^T \boldsymbol{x}_i \Phi(i,k) \right] \tilde{\boldsymbol{w}}_k. \end{aligned} \tag{6.B.2}$$

Assume that the sequence $\{\boldsymbol{x}_k\}$ is bounded as in (6.3) with $\mu B_x < 2$ and is PE with level $\gamma_0 > 0$. Let $N < \infty$ be such that $0 < \gamma(N) \leq \gamma_0$ (the sup in (6.7)

may not be achieved for a finite N). Note that even if the sup is not achieved by a finite N, by choosing N large enough, $\gamma(N)$ can be as close to γ_0 as necessary. We show in Lemma 6.B.1 below that

$$\tilde{\boldsymbol{w}}_k^T \left[\sum_{i=k}^{k+N-1} \Phi(i,k)^T \boldsymbol{x}_i^T \boldsymbol{x}_i \Phi(i,k) \right] \tilde{\boldsymbol{w}}_k \geq \frac{a(N)}{\left(1+\mu\sqrt{N}B_x\right)^2} \|\tilde{\boldsymbol{w}}_k\|^2 \quad \forall\, k \geq 0. \tag{6.B.3}$$

Using this result, it follows that

$$\|\tilde{\boldsymbol{w}}_{k+N}\|^2 - \|\tilde{\boldsymbol{w}}_k\|^2 \leq -\frac{(2-\mu B_x)a}{\left(1+\mu\sqrt{N}B_x\right)^2} \|\tilde{\boldsymbol{w}}_k\|^2 = -N\gamma \|\tilde{\boldsymbol{w}}_k\|^2, \tag{6.B.4}$$

which is the desired result. ♢

Lemma 6.B.1. *Inequality* (6.B.3) *holds under the conditions of Lemma 6.1.*

Proof: Define

$$\Phi_k^N \triangleq \sum_{i=k}^{k+N-1} \mu(i) \Phi(i,k)^T \boldsymbol{x}_i \boldsymbol{x}_i^T \Phi(i,k).$$

The idea is to find a lower bound for $\lambda_{\min}\left(\Phi_k^N\right)$. Let $\boldsymbol{\alpha}$ be a unit norm vector and define

$$c_i \triangleq \left| \sqrt{\mu(i)} \boldsymbol{x}_i^T \Phi(i,k) \boldsymbol{\alpha} \right|, \qquad \boldsymbol{\alpha}^T \Phi_k^N \boldsymbol{\alpha} = \sum_{i=k}^{k+N-1} c_i^2 \triangleq C^2.$$

Upper bounds for $\left| \sqrt{\mu(i)} \boldsymbol{x}_i^T \boldsymbol{\alpha} \right|$, $i = k \ldots k+N-1$ can be obtained, using the expansion of $\Phi(i,k)$,

$$\begin{aligned} \left| \sqrt{\mu(i)} \boldsymbol{x}_i^T \boldsymbol{\alpha} \right| \leq & \left| \sqrt{\mu(i)} \boldsymbol{x}_i^T \Phi(i,k) \boldsymbol{\alpha} \right| + \\ & + \left| \left(\sqrt{\mu(i)\mu(i-1)} \boldsymbol{x}_i^T \boldsymbol{x}_{i-1} \right) \left(\sqrt{\mu(i-1)} \boldsymbol{x}_{i-1}^T \Phi(i-1,k) \boldsymbol{\alpha} \right) + \ldots \right. \\ & \left. + \left(\sqrt{\mu(i)\mu} \boldsymbol{x}_i^T \boldsymbol{x}_k \right) \left(\sqrt{\mu} \boldsymbol{x}_k^T \boldsymbol{\alpha} \right) \right| \leq c_i + \mu B_x \left(c_{i-1} + \ldots + c_k \right). \end{aligned}$$

Define $d_i \triangleq \sum_{j=k}^{i-1} c_j$, $k \leq j \leq k+N-1$. From the PE condition, $\gamma(N) > 0$, and the inequalities above,

$$\begin{aligned} a(N) &\leq \sum_{i=k}^{k+N-1} \left(\sqrt{\mu(i)}\boldsymbol{x}_i^T\boldsymbol{\alpha}\right)^2 \leq \sum_{i=k}^{k+N-1} \left(c_i + \mu B_x d_i\right)^2 = \\ &= C^2 + 2\mu B_x \sum_{i=k}^{k+N-1} c_i d_i + \mu^2 B_x^2 \sum_{i=k}^{k+N-1} d_i^2. \end{aligned} \tag{6.B.5}$$

To evaluate this last quantity, note that the d_i can be written as (recall that $d_k = 0$)

$$\begin{bmatrix} d_{k+1} \\ \vdots \\ d_{k+N-1} \end{bmatrix} = \begin{bmatrix} c_k & 0 & \cdots & 0 \\ c_k & c_{k+1} & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ c_k & c_{k+1} & \cdots & c_{k+N-1} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \triangleq D_k \mathbf{1},$$

where

$$\mathbf{1} \triangleq \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^T.$$

Define also $\boldsymbol{c}_k = \begin{bmatrix} c_{k+1} & c_{k+2} & \cdots & c_{k+N-1} \end{bmatrix}^T$, then

$$\sum_{i=k}^{k+N-1} c_i d_i = \boldsymbol{c}_k^T D_k \mathbf{1} \leq \|\boldsymbol{c}_k\|_\infty \|D_k\|_\infty \|\mathbf{1}\|_\infty \leq \sqrt{N} C^2,$$

and

$$\sum_{i=k}^{k+N-1} d_i^2 = \mathbf{1}^T D_k^T D_k \mathbf{1} \leq \|D_k\|_\infty^2 \|\mathbf{1}\|_\infty^2 = N C^2,$$

where the fact that $\|\boldsymbol{c}_k\|_\infty \leq \|\boldsymbol{c}_k\|_2 = C$ was used. From these bounds and (6.B.5), it follows that

$$a(N) \leq C^2\left(1 + 2\mu B_x \sqrt{N} + \mu^2 B_x^2 N\right).$$

Since $\boldsymbol{\alpha}$ is any unit-norm vector, this implies that $a(N) \leq \lambda_{\min}\big(\Phi_k^N\big) \leq (1 + \mu\sqrt{N}B_x)^2C^2$, from which (6.B.3) follows.

With the exponential stability, it follows that the algorithm is also bounded-input bounded-state stable (BIBS), where here the noise $v(k)$ is the "input", and the weight-error vector $\tilde{\boldsymbol{w}}_k$ is the state (this fact, that exponential stability implies bounded-input bounded-output stability, is a standard result — see, for example, [Kha96, pp. 269–270]).

$\diamondsuit$

6.C PROOF OF THEOREM 6.1

To prove that the sequence $\big\{\tilde{\boldsymbol{z}}_k\big\}$ in (6.14) is bounded, introduce the Lyapunov function $V(\cdot,\cdot)$

$$V(k, \tilde{\boldsymbol{w}}_k) = \sum_{i=k}^{k+N-1} \|\tilde{\boldsymbol{z}}_i^{IP}\|^2. \tag{6.C.1}$$

In the definition of V, $\tilde{\boldsymbol{z}}_{k+i}^{IP}$ is the weight error vector that would be obtained if we use the *infinite precision* LMS algorithm *without noise* with the initial condition (at time k) $\tilde{\boldsymbol{z}}_k^{IP} = \tilde{\boldsymbol{z}}_k$. That is, $\tilde{\boldsymbol{z}}_i^{IP}$ is obtained from the recursion

$$\tilde{\boldsymbol{z}}_{i+1}^{IP} = \big(I - \mu(i)\boldsymbol{x}_i\boldsymbol{x}_i^T\big)\tilde{\boldsymbol{z}}_i^{IP}, \qquad \text{with initial condition } \tilde{\boldsymbol{z}}_k^{IP} = \tilde{\boldsymbol{z}}_k. \tag{6.C.2}$$

Next, some properties of V are derived. These properties are obtained by considering the noise-free IP LMS algorithm error vectors $\tilde{\boldsymbol{w}}_k$. Later these properties are used to analyze the FP noisy case. Let then $\tilde{\boldsymbol{w}}_k = \tilde{\boldsymbol{z}}_k$, and continue with the noise-free IP LMS algorithm for time $k+1, k+2, \ldots$:

$$\tilde{\boldsymbol{w}}_{i+1} = \big(I - \mu(i)\boldsymbol{x}_i\boldsymbol{x}_i^T\big)\tilde{\boldsymbol{w}}_i, \qquad \text{with initial condition } \tilde{\boldsymbol{w}}_k = \tilde{\boldsymbol{z}}_k. \tag{6.C.3}$$

In this case, $V(k+1, \tilde{\boldsymbol{w}}_{k+1}) - V(k, \tilde{\boldsymbol{w}}_k) = \|\tilde{\boldsymbol{w}}_{k+N}\|^2 - \|\tilde{\boldsymbol{w}}_k\|^2$. Hence, from Lemma 6.1, the one-step difference of the function V is bounded by the same quantity as on the right-hand side of (6.B.4).

$$V(k+1, \tilde{\boldsymbol{w}}_{k+1}) - V(k, \tilde{\boldsymbol{w}}_k) \leq -N\gamma \|\tilde{\boldsymbol{w}}_k\|^2. \tag{6.C.4}$$

The second property is obvious from the definition:

$$\|\boldsymbol{w}_k\|_2^2 \leq V(k, \tilde{\boldsymbol{w}}_k). \tag{6.C.5}$$

A sharper bound can be found if $2\mu B_x < 1$. In this case, from (6.5) it follows that

$$\|\tilde{\boldsymbol{w}}_{k+1}\|^2 \geq (1 - 2\mu B_x)\|\tilde{\boldsymbol{w}}_k\|^2.$$

Using this bound recursively in the definition of V,

$$\frac{1 - (1 - 2\mu B_x)^N}{2\mu B_x} \|\tilde{\boldsymbol{w}}_k\|^2 \leq V(k, \tilde{\boldsymbol{w}}_k). \tag{6.C.6}$$

Define B such that $V(k, \tilde{\boldsymbol{w}}_k) \geq B\|\tilde{\boldsymbol{w}}_k\|^2$, with B given by (6.C.5) or (6.C.6), depending on the values of N and μB_x.

An upper bound for V is found as follows. Under the assumption $\mu B_x < 2$ in (6.3), the matrices $\left(I - \mu(k)\boldsymbol{x}_k \boldsymbol{x}_k^T\right)$ have 2-norm bounded by 1, so for all $i, j \geq 0$, $i \geq j$, $\|\Phi(i,j)\|_2 \leq 1$ (see (6.B.1)), and

$$V(k, \tilde{\boldsymbol{w}}_k) \leq N\|\tilde{\boldsymbol{w}}_k\|_2^2. \tag{6.C.7}$$

The last inequality is used to bound the FP error terms. Let $\boldsymbol{y}_1$, $\boldsymbol{y}_2$ be any two M-vectors. Define $\left\{\tilde{\boldsymbol{w}}_i^l\right\}$ as the sequence obtained by applying recursion (6.C.3) with initial condition $\tilde{\boldsymbol{w}}_k = \boldsymbol{y}_l$, for $l = 1, 2$. Thus, the difference between $V(k,\, \boldsymbol{y}_1)$ and $V(k,\, \boldsymbol{y}_2)$ is

$$\left|V(k, \boldsymbol{y}_1) - V(k, \boldsymbol{y}_2)\right| = \left| \sum_{i=i}^{k+N-1} \left(\|\tilde{\boldsymbol{w}}_i^1\|^2 - \|\tilde{\boldsymbol{w}}_i^2\|^2 \right) \right|.$$

Using again the assumption $\mu B_x < 2$, $\|\tilde{\boldsymbol{w}}_i^l\| \leq \|\boldsymbol{y}_l\|$ $\forall i \geq k$, so

$$\big|V(k,\boldsymbol{y}_1) - V(k,\boldsymbol{y}_2)\big| \leq \Big|N\big(\|\boldsymbol{y}_1\|^2 - \|\boldsymbol{y}_2\|^2\big)\Big| \leq N\big(\|\boldsymbol{y}_1\| + \|\boldsymbol{y}_2\|\big)\|\boldsymbol{y}_1 - \boldsymbol{y}_2\|. \tag{6.C.8}$$

Now these properties are extended to the FP, noisy case. Remark that the first term on the RHS of LMS's error equation (6.14) is exactly $\tilde{\boldsymbol{z}}_{k+1}^{IP}$ as given by (6.C.2). That is,

$$\begin{aligned}\tilde{\boldsymbol{z}}_{k+1} = \tilde{\boldsymbol{z}}_{k+1}^{IP} - \mu(k)\boldsymbol{x}_k v(k) - \operatorname{diag}\big(\delta_{1,l}\big)\boldsymbol{z}_k - \operatorname{diag}\big(\delta_{2,l}\big)\mu(k)\check{e}(k)\boldsymbol{x}_k - \\ - \operatorname{diag}\big(\delta_{3,l}\big)\mu(k)\|\boldsymbol{x}_k\|\|\boldsymbol{z}_k\|\boldsymbol{x}_k,\end{aligned} \tag{6.C.9}$$

Assume that the regressor sequence $\big\{\mu(k)\boldsymbol{x}_k\big\}$ is PE with level $\gamma_0 > 0$ and satisfies $\mu B_x < 2$. Again, one can choose a finite N such that $\gamma(N)$ is as close to γ_0 as desired. In this case, (6.C.8) and (6.C.9) can be used to obtain

$$\begin{aligned}V(k+1,\tilde{\boldsymbol{z}}_{k+1}) - V(k,\tilde{\boldsymbol{z}}_k) &\leq V(k+1,\tilde{\boldsymbol{z}}_{k+1}^{IP}) - V(k,\tilde{\boldsymbol{z}}_k) + \\ &+ \big|V(k+1,\tilde{\boldsymbol{z}}_{k+1}) - V(k+1,\tilde{\boldsymbol{z}}_{k+1}^{IP})\big| \leq \\ &\leq -N\gamma\|\tilde{\boldsymbol{z}}_k\|^2 + N\Big(\|\tilde{\boldsymbol{z}}_{k+1}\| + \|\tilde{\boldsymbol{z}}_{k+1}^{IP}\|\Big)\|\tilde{\boldsymbol{z}}_{k+1} - \tilde{\boldsymbol{z}}_{k+1}^{IP}\|,\end{aligned}$$

where the difference $V(k,\tilde{\boldsymbol{z}}_{k+1}^{IP}) - V(k,\tilde{\boldsymbol{z}}_k)$ was bounded using (6.C.4). The norms in the above inequality can be evaluated as follows:

$$\|\tilde{\boldsymbol{z}}_{k+1}^{IP}\| = \Big\|\Big(I - \mu(k)\boldsymbol{x}_k\boldsymbol{x}_k\Big)\tilde{\boldsymbol{z}}_k\Big\| \leq \|\tilde{\boldsymbol{z}}_k\|,$$

$$\begin{aligned}\|\tilde{\boldsymbol{z}}_{k+1}\| \leq \Big[1 + \delta_1 + \big(\delta_2 + \delta_3\big)\mu(k)\|\boldsymbol{x}_k\|^2\Big]\|\tilde{\boldsymbol{z}}_k\| + \\ + \big(1 + \delta_2\big)\mu(k)\|\boldsymbol{x}_k\|\,|v(k)| + \big(\delta_1 + \delta_3\mu(k)\|\boldsymbol{x}_k\|^2\big)\|\boldsymbol{w}\|,\end{aligned}$$

and

$$\begin{aligned}\|\tilde{\boldsymbol{z}}_{k+1} - \tilde{\boldsymbol{z}}_{k+1}^{IP}\| \leq \Big[\delta_1 + \big(\delta_2 + \delta_3\big)\mu(k)\|\boldsymbol{x}_k\|^2\Big]\|\tilde{\boldsymbol{z}}_k\| + \\ + \big(1 + \delta_2\big)\mu(k)\|\boldsymbol{x}_k\|\,|v(k)| + \big(\delta_1 + \delta_3\mu(k)\|\boldsymbol{x}_k\|^2\big)\|\boldsymbol{w}\|.\end{aligned}$$

Using these relations,

$$\Big(\|\tilde{\boldsymbol{z}}_{k+1}\| + \|\tilde{\boldsymbol{z}}^{IP}_{k+1}\|\Big)\|\tilde{\boldsymbol{z}}_{k+1} - \tilde{\boldsymbol{z}}^{IP}_{k+1}\| \leq \delta_4(1+\delta_4)\|\tilde{\boldsymbol{z}}_k\|^2 +$$
$$+ \delta_4(1+\delta_2)\mu(k)\|\boldsymbol{x}_k\|\,\|\tilde{\boldsymbol{z}}_k\|\,|v(k)| + +\delta_4\delta_5\|\tilde{\boldsymbol{z}}_k\|\|\boldsymbol{w}\| +$$
$$+ (1+\delta_2)(1+\delta_4)\mu(k)\|\boldsymbol{x}_k\|\,\|\tilde{\boldsymbol{z}}_k\|\,|v(k)| + (1+\delta_2)^2\mu^2(k)\|\boldsymbol{x}_k\|^2 v^2(k) +$$
$$+ \delta_5(1+\delta_4)\|\tilde{\boldsymbol{z}}_k\|\,\|\boldsymbol{w}\| + 2\delta_5(1+\delta_2)\mu(k)\|\boldsymbol{x}_k\|\,\|\boldsymbol{w}\|\,|v(k)| + \delta_5^2\|\boldsymbol{w}\|^2, \tag{6.C.10}$$

where $\delta_4 = \delta_1 + \mu B_x(\delta_2 + \delta_3)$ and $\delta_5 = \delta_1 + \mu B_x \delta_3$.

Applying Young's inequality (see Appendix 6.A) to the cross-terms in (6.C.10) and regrouping,

$$V(k+1, \tilde{\boldsymbol{z}}_{k+1}) - V(k, \tilde{\boldsymbol{z}}_k) \leq -N\Big(\gamma - \rho\big(1+\xi_1\big) - \xi_2\Big)\|\tilde{\boldsymbol{z}}_k\|^2 +$$
$$+ N\left(\frac{1}{\rho}\xi_3 + \xi_4\right)\|\boldsymbol{w}\|^2 + N\left(\frac{1}{\rho}\big(1+\xi_5\big) + 1 + \xi_6\right)\mu B_x v^2(k). \tag{6.C.11}$$

The factor ρ is used to bound the cross-terms containing $\|\tilde{\boldsymbol{z}}_k\|$ in (6.C.10) and will be chosen shortly. The expressions for the ξ_i can (with some patience) be computed from the definitions for the δ_i. The approximations below assume that the mantissa of a floating-point number is stored with at least 5 bits, so that $\varepsilon < 3.125 \times 10^{-2}$:

$$\xi_1 \leq 2.1\varepsilon + 9.4\mu B_x\varepsilon + 3.2\mu B_x M\varepsilon + 1.1\mu^2 B_x^2 M^2\varepsilon^2$$
$$\xi_2 \leq 6.3\varepsilon + 5.1\mu B_x\varepsilon + 2.6\mu B_x M\varepsilon + 1.1\mu^2 B_x^2 M^2\varepsilon^2$$

$$\xi_3 \leq 1.4\varepsilon + 1.5\mu B_x M\varepsilon + 1.1\mu^2 B_x^2 M^2\varepsilon^2 \quad \xi_4 \leq 1.2\varepsilon + 1.5\mu B_x M\varepsilon + 1.1\mu^2 B_x^2 M^2\varepsilon^2$$
$$\xi_5 \leq 5.3\varepsilon + 5.1\mu B_x\varepsilon + 1.4\mu B_x M\varepsilon \qquad \xi_6 \leq 10\varepsilon + 1.5\mu B_x M\varepsilon$$

The positive scalar ρ must be chosen so that the coefficient of $\|\tilde{\boldsymbol{z}}_k\|^2$ in (6.C.11) is strictly negative. It can be seen that this is possible if $\gamma > \xi_2$. If this condition

is satisfied, then the one step difference $V(k+1, \tilde{\boldsymbol{z}}_{k+1}) - V(k, \tilde{\boldsymbol{z}}_k)$ will be negative whenever

$$\|\tilde{\boldsymbol{z}}_k\|^2 > V_0 \triangleq \frac{\left(\frac{1}{\rho}\xi_3 + \xi_4\right)\|\boldsymbol{w}\|^2 + \left(\frac{1}{\rho}\big(1+\xi_5\big) + 1 + \xi_6\right)\mu B_x v_{\max}^2}{\gamma - \rho\big(1+\xi_1\big) - \xi_2}. \qquad (6.C.12)$$

Using this fact, the results follow directly from standard Lyapunov stability results [Kha96, Ch. 3] and the properties of V listed above. Recall that $V(k, \tilde{\boldsymbol{z}}_k) \leq N\|\tilde{\boldsymbol{z}}_k\|^2$, so $V(k+1, \tilde{\boldsymbol{z}}_{k+1})$ will be strictly smaller than $V(k, \tilde{\boldsymbol{z}}_k)$ whenever $V(k, \tilde{\boldsymbol{z}}_k)$ is larger than NV_0. This implies that for large k, $\tilde{\boldsymbol{z}}_k$ will stay in the region $\mathcal{F}$ defined by

$$\mathcal{F} \triangleq \Big\{\tilde{\boldsymbol{z}}_k \;:\; V(k, \tilde{\boldsymbol{z}}_k) \leq NV_0\Big\}.$$

The region $\mathcal{F}$ is defined in terms of V. To obtain a description in terms of $\|\tilde{\boldsymbol{z}}_k\|$, invoke the relation $B\|\tilde{\boldsymbol{z}}_k\|^2 \leq V(k, \tilde{\boldsymbol{z}}_k)$. Using this relation, one can show that $\mathcal{F}$ is contained in the ball

$$\mathcal{G} \triangleq \Big\{\tilde{\boldsymbol{z}}_k \;:\; \|\tilde{\boldsymbol{z}}_k\|^2 \leq \frac{N}{B}V_0\Big\}.$$

We then conclude that for large k, $\tilde{\boldsymbol{z}}_k$ will stay inside $\mathcal{G}$, which is exactly the bound given in the text if $C = \frac{N}{B}$.

$\diamondsuit$

6.D FINITE-PRECISION UPDATE LAWS

We assume, as explained in Sec. 6.1.2, that all algorithms are implemented using fixed-point arithmetic, where all variables are stored with B bits plus sign, and that rounding is used (with $\varepsilon = 2^{-B-1}$). We also assume that $\boldsymbol{x}_k$ and $y(k)$ represent already quantized variables (i.e., there are exact fixed-point representations for $y(k)$ and $\boldsymbol{x}_k$).

In fixed-point arithmetic, additions are performed without error if the variables are scaled so that overflow does not occur. On the other hand, there is an error when performing a multiplication, say $\mathrm{fx}[ab] = ab + \delta$, where $|\delta| \leq \varepsilon$. One usually assumes that δ is a random variable with uniform distribution and zero mean (so that its variance is $\sigma_d^2 = \frac{2^{-2B}}{12}$), and that δ is independent of both a and b. It is also common to assume that errors in two different operations are independent. Note that none of these assumptions is exactly true — in particular, there are systems in which $\mathrm{E}\,\delta$, though small, is nonzero (see the discussion in Sec. 6.1.2).

To differentiate between the infinite and finite-precision versions of the various algorithms, the weight estimates computed by the fixed-point algorithms are denoted by $\boldsymbol{z}_k$ (for LMS), $\boldsymbol{z}_k^c$ (for circular-leaky), and $\boldsymbol{z}_k^s$ (for switching-σ). Similarly, the weight error vectors are $\tilde{\boldsymbol{z}}_k$, $\tilde{\boldsymbol{z}}_k^c$, and $\tilde{\boldsymbol{z}}_k^s$.

Circular-leaky. In fixed-point, the update law of circular-leaky is given by

$$\boldsymbol{z}_{k+1}^c = \mathrm{fx}\left[\left(I - \mu\alpha_c(z_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\right)\boldsymbol{z}_k^c\right] + \mathrm{fx}\left[\mu\boldsymbol{x}_k\,\mathrm{fx}\left[e^c(k)\right]\right], \tag{6.D.1}$$

where $e^c(k) = \boldsymbol{z}_k^T\boldsymbol{x}_k$. We now expand the terms $\mathrm{fx}[\cdot]$ in (6.D.1), starting with $e_Q^c(k) \stackrel{\Delta}{=} \mathrm{fx}[e^c(k)]$. Following [Wil63], we obtain

$$e_Q^c(k) = \mathrm{fx}[\boldsymbol{z}_k^T\boldsymbol{x}_k] = \boldsymbol{z}_k^T\boldsymbol{x}_k + \eta(k),$$

where the error $\eta(k)$ satisfies[7]

$$|\eta(k)| \leq M\varepsilon, \qquad\qquad \mathrm{E}\,\eta(k)^2 \stackrel{\Delta}{=} \sigma_\eta^2 = M\sigma_d^2.$$

Similarly, define the error $\boldsymbol{\xi}_k$ by

$$\boldsymbol{\xi}_k \stackrel{\Delta}{=} \mathrm{fx}[\mu\boldsymbol{x}_k e_Q^c(k)] - \mu\boldsymbol{x}_k e_Q^c(k).$$

[7] If the multiplications are computed in double precision, and only the final result is rounded to B bits, then $|\eta(k)| \leq \varepsilon$ and $\sigma_\eta^2 = \sigma_d^2$.

Expanding the left-hand side, we obtain

$$\mathrm{fx}[\mu \boldsymbol{x}_k e_Q^c(k)] = \mathrm{fx}\Big[\mathrm{fx}[\mu e_Q^c(k)]\boldsymbol{x}_k\Big] = \big(\mu e_Q^c(k) + \xi_k^{'}\big)\boldsymbol{x}_k + \boldsymbol{\xi}_k^{''},$$

where

$$\begin{aligned} |\xi_k^{'}| &\le \varepsilon, & \mathrm{E}\,\xi_k^{'2} &= \sigma_d^2, \\ \|\boldsymbol{\xi}_k^{''}\| &\le \sqrt{M}\varepsilon, & \mathrm{E}\,\boldsymbol{\xi}_k^{''}\boldsymbol{\xi}_k^{''\,T} &= \sigma_d^2 I, \end{aligned}$$

from which we conclude that

$$\|\boldsymbol{\xi}_k\| \le \sqrt{M}\varepsilon + \varepsilon\|\boldsymbol{x}_k\|, \qquad \mathrm{E}\,\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T \triangleq \Sigma_\xi = \sigma_d^2\big(I + R\big).$$

The last term we need to evaluate is $\mathrm{fx}\big[(I - \mu\alpha_c \boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c\big]$. If $\alpha_c(z_{k,\,\bar{k}}^c) = 0$, $\boldsymbol{z}_k^c$ is not modified, and there is no error. On the other hand, if $|z_{k,\,\bar{k}}^c| \ge C_1$, we have

$$\mathrm{fx}\big[(I - \mu\alpha_c \boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c\big] = (I - \mu\rho\alpha_0 \boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c + \zeta(k)\boldsymbol{e}_{\bar{k}},$$

where $|\zeta(k)| \le \varepsilon$, $\mathrm{E}\,\zeta(k)^2 = \sigma_d^2$, and

$$\rho = \begin{cases} 1, & \text{if } \big|z_{k,\,\bar{k}}\big| \ge C_2, \\ \frac{1}{\mu\alpha_0}\,\mathrm{fx}\left[\mu\alpha_c\big(z_{k,\,\bar{k}}\big)\right], & \text{if } C_1 < \big|z_{k,\,\bar{k}}\big| < C_2. \end{cases}$$

In general, it can be shown that ρ satisfies $0 \le \rho < 1 + 3\varepsilon$. However, it turns out that the error incurred in computing ρ does not affect our analysis in an important way, so in the following we will assume that this error is zero.

The combination of all finite-precision errors is denoted by $\boldsymbol{\delta}_k^c$, i.e.,

$$\boldsymbol{\delta}_k^c \triangleq \boldsymbol{\xi}_k + \zeta(k)\boldsymbol{e}_{\bar{k}} + \mu\eta(k)\boldsymbol{x}_k.$$

From our assumptions, it follows that $\boldsymbol{\delta}_k^c$ satisfies

$$\|\boldsymbol{\delta}_k^c\| \le \Big(\sqrt{M} + (1 + \mu\sqrt{M})\|\boldsymbol{x}_k\| + 1(\alpha_c \ne 0)\Big)\varepsilon\,, \tag{6.D.2}$$

$$\mathrm{E}\,\boldsymbol{\delta}_k^c\boldsymbol{\delta}_k^{c\,T} = \Sigma_\xi + \mu\sigma_d^2 R + 1(\alpha_c \ne 0)\sigma_d^2 \boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\,, \tag{6.D.3}$$

$$|\delta_{k,\,j}^c| \le \Big(2 + (1 + \mu)\|\boldsymbol{x}_k\|_\infty\Big)\varepsilon\,, \tag{6.D.4}$$

where $1(\alpha_c \neq 0) = 1$ if $\alpha_c \neq 0$ and zero otherwise. Also, the last equation provides bounds for the individual elements of $\boldsymbol{\delta}_k^c$.

With these definitions, we can write the finite-precision update law for circular leaky as

$$\boldsymbol{z}_{k+1}^c = \boldsymbol{z}_k + \mu \boldsymbol{x}_k e^c(k) - \mu \alpha_c \boldsymbol{e}_{\bar{k}} \boldsymbol{e}_{\bar{k}}^T \boldsymbol{z}_k + \boldsymbol{\delta}_k^c, \tag{6.D.5}$$

Modified Switching-σ. The update law for the switching-σ algorithm is obtained by following a similar procedure. The result is

$$\boldsymbol{z}_{k+1}^s = \Big((1 - \mu\alpha_s(\boldsymbol{z}_k^s))I - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T\Big)\boldsymbol{z}_k^s + \mu \boldsymbol{x}_k \boldsymbol{x}_k^T \boldsymbol{w}_* + \mu \boldsymbol{x}_k v(k) + \boldsymbol{\delta}_k^s. \tag{6.D.6}$$

The only difference is in the term $\boldsymbol{\delta}_k^s$, which satisfies

$$\begin{aligned}
\|\boldsymbol{\delta}_k^s\| &\leq \Big(\sqrt{M} + (1 + \mu\sqrt{M})\|\boldsymbol{x}_k\| + 1(\alpha_s \neq 0)\sqrt{M}\Big)\varepsilon \ , \\
\mathrm{E}\, \boldsymbol{\delta}_k^s \boldsymbol{\delta}_k^{s\,T} &= \Sigma_\xi + \mu\sigma_d^2 R + 1(\alpha_s \neq 0)\sigma_d^2 I \ , \\
|\delta_{k,\,j}^s| &\leq \Big(1 + (1+\mu)\|\boldsymbol{x}_k\|_\infty + 1(\alpha_s \neq 0)\Big)\varepsilon.
\end{aligned}$$

6.E AVERAGED SYSTEM FOR SWITCHING-σ

We need to show that in steady-state, the averaged variable $\boldsymbol{z}_k^{s,\,av}$ has norm less than S_1, so that the leakage term remains equal to zero. To do so, we compare the averaged recursion (6.44) with the averaged LMS recursion

$$\boldsymbol{z}_{k+1}^{av} = \big(I - \mu R\big)\boldsymbol{z}_k^{av} + \mu R \boldsymbol{w}_*.$$

Note that, if $\big|\lambda(I - \mu R)\big| < 1$, this recursion satisfies

$$\lim_{k\to\infty} \boldsymbol{z}_k^{av} = \boldsymbol{w}_*. \tag{6.E.1}$$

On the other hand, expanding $\boldsymbol{z}_{k+1}^{av}$, we obtain

$$\boldsymbol{z}_{k+1}^{av} = \prod_{i=0}^{k} \big[I - \mu R\big] \boldsymbol{z}_0^{av} + \sum_{i=0}^{k} \prod_{j=i+1}^{k} \big[I - \mu R\big] \mu R \boldsymbol{w}_*.$$

Since the first term in the above relation tends to zero, (6.E.1) implies that

$$\lim_{k\to\infty} \left(\sum_{i=0}^{k} \prod_{j=i+1}^{k} \big[I - \mu R\big] \right) \mu R = I,$$

and therefore the relation below holds for any vector $\boldsymbol{y}$ with unit norm

$$\lim_{k\to\infty} \boldsymbol{y}^T \left(\sum_{i=0}^{k} \prod_{j=i+1}^{k} \big[I - \mu R\big] \right) \mu R \boldsymbol{y} = 1. \tag{6.E.2}$$

We will now rewrite this expression in a more adequate form. Let $R^{1/2}$ be a symmetric square-root factor of R (i.e., $\left(R^{1/2}\right)^2 = R$, $\left(R^{1/2}\right)^T = R^{1/2}$). We can rewrite each term in the above sum as below.

$$\left(\prod_{j=i+1}^{k} \big[I - \mu R\big] \right) \mu R = \mu R^{1/2} \left(\prod_{j=i+1}^{k} \big[I - \mu R\big] \right) R^{1/2}, \tag{6.E.3}$$

which follows from

$$(I - \mu R)R = (R - \mu R^2) = (R^{1/2}R^{1/2} - \mu R^{1/2}RR^{1/2}) = R^{1/2}(I - \mu R)R^{1/2}$$

and

$$(I - \mu R)R^{1/2} = (R^{1/2} - \mu R^{1/2}R) = R^{1/2}(I - \mu R).$$

From (6.E.3) and (6.E.2), we obtain

$$\lim_{k\to\infty} \mu \sum_{i=0}^{k} \boldsymbol{y}^T R^{1/2} \left(\prod_{j=i+1}^{k} \big[I - \mu R\big] \right) R^{1/2} \boldsymbol{y} \triangleq \mu \sum_{i=0}^{k} \boldsymbol{y}^T B_i \boldsymbol{y} = 1, \tag{6.E.4}$$

where we have defined the matrices B_i. Assuming that $0 < \mu R < I$, all the B_i are positive-definite, and thus all the terms in the above sum are positive. We shall use this result soon.

Returning to the switching-σ algorithm, we have

$$\boldsymbol{z}_{k+1}^{s,\,av} = \prod_{i=0}^{k} \big[(1-\mu\alpha_s)I - \mu R\big]\boldsymbol{z}_0^{s,\,av} + \sum_{i=0}^{k}\prod_{j=i+1}^{k}\big[(1-\mu\alpha_s)I-\mu R\big]\mu R\boldsymbol{w}_*.$$

If $-1 < 1-\mu\alpha_0 - \lambda_i(R) < 1$, the term in $\boldsymbol{z}_0^{s,\,av}$ will tend to zero, and therefore

$$\lim_{k\to\infty}\boldsymbol{z}_{k+1}^{s,\,av} = \lim_{k\to\infty}\sum_{i=0}^{k}\prod_{j=i+1}^{k}\big[(1-\alpha_s)I-\mu R\big]\mu R\boldsymbol{w}_*. \tag{6.E.5}$$

Taking norms on both sides, we obtain

$$\|\tilde{\boldsymbol{z}}_{\infty}^{av}\| \le \left\|\sum_{i=0}^{k}\prod_{j=i+1}^{k}\big[(1-\alpha_s)I-\mu R\big]\mu R\right\|\,\|\boldsymbol{w}_*\|.$$

We will now use (6.E.4) to show that the above matrix norm is no greater than one, from which we can conclude that $\|\tilde{\boldsymbol{z}}_k^{av}\| \le \|\boldsymbol{w}_*\| < S_1$ for large enough k.

Similarly to what we did in (6.E.3), we can write

$$\prod_{j=i+1}^{k}\big[(1-\mu\alpha_s)I-\mu R\big]\mu R = \mu R^{1/2}\left(\prod_{j=i+1}^{k}\big[(1-\mu\alpha_s)I-\mu R\big]\right)R^{1/2}.$$

Assume that $1-\mu\lambda_{\max}(R) > 0$ and choose α_0 such that

$$|1-\mu\alpha_0-\mu\lambda_i(R)| < 1-\mu\lambda_i(R) < 1, \quad \text{that is,} \quad 0<\alpha_0<2-2\mu\lambda_{\max}(R). \tag{6.E.6}$$

If this condition is satisfied, then for all k,

$$-B_i \le \mu R^{1/2}\left(\prod_{j=i+1}^{k}\big[(1-\mu\alpha_s)I-\mu R\big]\right)R^{1/2} \le B_i.$$

Let $\boldsymbol{y}$ be a unit-norm vector. From the above relations we conclude that

$$\boldsymbol{y}^T B_i \boldsymbol{y} \ge \left|\mu\boldsymbol{y}^T R^{1/2}\left(\prod_{j=i+1}^{k}\big[(1-\mu\alpha_s)I-\mu R\big]\right)R^{1/2}\boldsymbol{y}\right|.$$

Therefore, we have the bound

$$\left\|\sum_{i=0}^{\infty}\prod_{j=i+1}^{\infty}\left[(1-\mu\alpha_s)I-\mu R\right]\mu R\right\| =$$
$$= \max_{\|\boldsymbol{y}\|=1}\left|\sum_{i=0}^{\infty}\mu\boldsymbol{y}^T R^{1/2}\left(\prod_{j=i+1}^{\infty}\left[(1-\mu\alpha_s)I-\mu R\right]\right)R^{1/2}\boldsymbol{y}\right| \leq$$
$$\leq \max_{\|\boldsymbol{y}\|=1}\sum_{i=0}^{\infty}\boldsymbol{y}^T B_i\boldsymbol{y} = 1.$$

This relation and (6.E.5) imply that there is a K such that, for all $k \geq K$,

$$\lim_{k\to\infty}\|\boldsymbol{z}_k^{s,\,av}\| \leq \|\boldsymbol{w}_*\| < S_1,$$

completing our proof.

6.F PROOF OF THEOREM 6.5

The variable $\boldsymbol{z}_{k+M}^c$ can be shown to satisfy

$$\boldsymbol{z}_{k+M}^c = \prod_{l=k}^{k+M-1}\left(I-\mu\alpha_c\boldsymbol{e}_{\bar{l}}\boldsymbol{e}_{\bar{l}}^T-\mu\boldsymbol{x}_l\boldsymbol{x}_l^T\right)\boldsymbol{z}_k^c +$$
$$+\sum_{i=k}^{k+M-1}\left[\prod_{l=i}^{k+M-1}\left(I-\mu\alpha_c\boldsymbol{e}_{\bar{l}}\boldsymbol{e}_{\bar{l}}^T-\mu\boldsymbol{x}_l\boldsymbol{x}_l^T\right)\left(\mu\boldsymbol{x}_i\boldsymbol{x}_i^T\boldsymbol{w}_*-\mu\boldsymbol{x}_i v(i)-\boldsymbol{\delta}_i^c\right)\right]. \tag{6.F.1}$$

Given that α_0 and μ satisfy $|1-\mu\alpha_0-\mu B_x| < 1$, we find that all matrices in the expression for $\boldsymbol{z}_{k+M}^c$ above are contractive (i.e., have 2-induced norms less than or equal to 1). It follows that the second term in (6.F.1) is bounded by

$$\sum_{i=k}^{k+M-1}\left[\prod_{l=i}^{k+M-1}\left(I-\mu\alpha_c\boldsymbol{e}_{\bar{l}}\boldsymbol{e}_{\bar{l}}^T-\mu\boldsymbol{x}_l\boldsymbol{x}_l^T\right)\left(\mu\boldsymbol{x}_i\boldsymbol{x}_i^T\boldsymbol{w}_*-\mu\boldsymbol{x}_i v(i)-\boldsymbol{\delta}_i^c\right)\right] \leq$$
$$\leq M\left[\mu B_x\|\boldsymbol{w}_*\|+\sqrt{\mu B_x}v_{\max}+\|\boldsymbol{\delta}_i^c\|\right]. \tag{6.F.2}$$

We also need to bound the norm of the first term of (6.F.1). Define

$$A_i = I - \mu\alpha_c \boldsymbol{e}_{\bar{i}}\boldsymbol{e}_{\bar{i}}^T, \qquad\qquad B_i = \mu \boldsymbol{x}_i \boldsymbol{x}_i^T,$$

and note that $\|B_i\| \leq \mu B_x$. In addition, if $|1 - \mu\alpha_0 - \mu B_x| < 1$ we have

$$\|A_i - B_i\| \leq 1. \tag{6.F.3}$$

With these definitions, the product we want to bound is

$$\prod_{i=k}^{k+M-1}\Big(A_i - B_i\Big) = \prod_{i=k}^{k+M-1} A_i - \sum_{i=k}^{k+M-1}\left[\prod_{j=i+1}^{k+M-1}\big(A_j - B_j\big)\right] B_i \left[\prod_{l=k}^{i-1}\big(A_l - B_l\big)\right].$$

Consider the second term. From (6.F.3), its norm is bounded by

$$\left\|\sum_{i=k}^{k+M-1}\left[\prod_{j=i+1}^{k+M-1}\big(A_j - B_j\big)\right] B_i \left[\prod_{l=k}^{i-1}\big(A_l - B_l\big)\right]\right\| \leq \sum_{i=k}^{k+M-1}\|B_i\| \leq M\mu B_x.$$

To approximate the first term, note that $\boldsymbol{e}_{\bar{i}}^T\boldsymbol{e}_{\bar{j}} = 0$ if $0 < |i-j| < M$, thus

$$\prod_{i=k}^{k+M-1} A_i = I - \sum_{i=k}^{k+M-1}\boldsymbol{e}_{\bar{i}}\boldsymbol{e}_{\bar{i}}^T.$$

Now let $z^c_{k,\,m}$ denote the entry of $\boldsymbol{z}^c_k$ that has the largest absolute value ($|z^c_{k,\,m}| = \|\boldsymbol{z}^c_k\|_\infty$). Let $k+l$ be such that $\overline{k+l} = m$ and assume for now that $\alpha_c(z^c_{k+l,m}) = \alpha_0$, so that

$$\left\|\Big[I - \sum_{i=k}^{k+M-1}\alpha_c(z^c_{i,\,\bar{i}})\boldsymbol{e}_{\bar{i}}\boldsymbol{e}_{\bar{i}}^T\Big]\boldsymbol{z}^c_k\right\| = \left\|\boldsymbol{z}^c_k - \sum_{i=k}^{k+M-1}\alpha_c(z^c_{i,\,\bar{i}})\boldsymbol{e}_{\bar{i}}z^c_{k,\,\bar{i}}\right\| \leq$$

$$\leq \left[\sum_{\substack{i=1\\ i\neq m}}^{M} {z^c_{k,\,i}}^2 + (1-\alpha_0)^2 {z^c_{k,\,m}}^2\right]^{1/2} \leq \tag{6.F.4}$$

$$\leq \sqrt{1 - \frac{\alpha_0(2-\alpha_0)}{M}}\,\|\boldsymbol{z}^c_k\|. \tag{6.F.5}$$

Putting all these results together, we obtain

$$\left\|\prod_{i=k}^{k+M-1}\Big(A_i - B_i\Big)\boldsymbol{z}^c_k\right\| \leq \left[\sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + M\mu B_x\right]\|\boldsymbol{z}^c_k\|. \tag{6.F.6}$$

Assume that $\alpha_0 > B_x$, such that

$$\sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + M\mu B_x < 1.$$

We still need to show that if $\|\boldsymbol{z}_k^c\|$ is large enough, then $|z_{k+l,\,m}^c| > C_2$ and $\alpha_c(z_{k+l,m}^c) = \alpha_0$. The expression for $z_{k+l,\,m}^c = \boldsymbol{e}_m^T \boldsymbol{z}_{k+l}^c$ is

$$z_{k+l,\,m}^c = \boldsymbol{e}_m^T \prod_{l=k}^{k+l-1} \left(I - \mu\alpha_c \boldsymbol{e}_{\bar{l}} \boldsymbol{e}_{\bar{l}}^T - \mu \boldsymbol{x}_l \boldsymbol{x}_l^T \right) \boldsymbol{z}_k^c + \\ + \boldsymbol{e}_m^T \sum_{i=k}^{k+l-1} \left[\prod_{j=i}^{k+l-1} \left(I - \mu\alpha_c \boldsymbol{e}_{\bar{j}} \boldsymbol{e}_{\bar{j}}^T - \mu \boldsymbol{x}_j \boldsymbol{x}_j^T \right) \left(\mu \boldsymbol{x}_i \boldsymbol{x}_i^T \boldsymbol{w}_* - \mu \boldsymbol{x}_i v(i) - \boldsymbol{\delta}_i^c \right) \right],$$

Note that $\boldsymbol{e}_m^T \boldsymbol{e}_{\bar{i}} = 0$ for $k \le i \le k+l-1$, and so, using again the decomposition of $\prod(A_i - B_i)$, we conclude that

$$|z_{k+l,\,m}^c| \ge |z_{k,\,m}^c| - M\mu B_x \|\boldsymbol{z}_k^c\| - M\left(\mu B_x \|\boldsymbol{w}_*\| + \sqrt{\mu B_x} v_{\max} + \|\delta^c\|\right) > C_2 \quad \text{if}$$

$$\|\boldsymbol{z}_k^c\|_\infty = |z_{k,\,m}^c| > C_2 + \mu M B_x \|\boldsymbol{z}_k^c\| + \\ + M\left(\mu B_x \|\boldsymbol{w}_*\| + \sqrt{\mu B_x} v_{\max} + \left(\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1 \right)\varepsilon \right).$$

Since $\|\boldsymbol{z}_k^c\| \le \sqrt{M}\|\boldsymbol{z}_k^c\|_\infty$, it follows from the above inequality that the leakage term will be equal to α_0 at time $k+l$ (where $\overline{k+l} = m$) if

$$\|\boldsymbol{z}_k^c\| \ge \frac{\sqrt{M}\left(C_2 + M\left(\mu B_x \|\boldsymbol{w}_*\| + \sqrt{\mu B_x}\, v_{\max} + \left(\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1 \right)\varepsilon \right) \right)}{1 - \mu M^{3/2} B_x} \triangleq \Omega_1^c.$$

If the above condition holds, using (6.F.6) and (6.F.2) we conclude that the norm $\|\boldsymbol{z}_{k+M}^c\|$ will be smaller than $\|\boldsymbol{z}_k^c\|$ if

$$\|\boldsymbol{z}_k^c\| \ge \max\left\{ \Omega_1^c, \quad M \frac{\mu B_x \|\boldsymbol{w}_*\| + \sqrt{\mu B_x}\, v_{\max} + \left(\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1 \right)\varepsilon}{1 - \sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + \mu M B_x} \right\} \triangleq \Omega_2^c$$

Therefore, $\|\boldsymbol{z}_{k+M}^c\|$ will be strictly smaller than $\|\boldsymbol{z}_k^c\|$ if $\|\boldsymbol{z}_k^c\| > \Omega_2^c$. From this point we can use an argument similar to that of Theorem 6.4 to show that the sequence $\left\{\|\boldsymbol{z}_k^c\|\right\}_{k=0}^\infty$ is bounded.

$\diamondsuit$

CHAPTER 7

SUGGESTIONS FOR FUTURE RESEARCH

This dissertation studied several aspects of the performance of adaptive filters without slow adaptation approximations. As we saw in the earlier chapters, the methods of analysis and design tend to be considerably more complex and more demanding in this case.

Despite the encouraging progress that has been made in this work, there are of course a variety of open issues that deserve further investigation and a closer study. We list below some suggestions for further work in this area.

Chapter 3

1. *Computing the MSE for NLMS.*

   We did not provide an expression for the MSE of the normalized LMS algorithm in Sec. 3.4 because there is no simple relation between $\mathrm{E}\, e(k)^2$ and the the averaged normalized error, $\mathrm{E}\, e^{(n)}(k)^2$, and because Theorem 3.4 only evaluates the MSD. Although Theorem 3.5 computes the MSE, its proof is specific for LMS. An extension of Theorem 3.4 to compute the MSE for NLMS and even other update laws would be useful.

Chapter 4

1. *Improving the stability bound* (4.32).

A tighter bound for the largest eigenvalue of the state-space matrix Φ would allow us to compute the step-size that provides the fastest convergence rate (in the mean-square sense), since

$$\text{convergence rate of dominant mode } = \rho(\Phi),$$

where $\rho(\Phi)$ is the spectral radius of Φ. A good approximation for $\rho(\Phi)$ thus gives an approximation for the dominant mode of the linear system

$$\boldsymbol{\Gamma}_{k+1} = \Phi \boldsymbol{\Gamma}_k$$

that describes the mean-square behavior of LMS.

An improved bound could be sought, for example, by applying a third set of similarity transformations to Φ, in order to eliminate the $O(\mu^3)$ terms in the recursions for some seed variables.

2. *Extending the state-space analysis to more general input and noise statistics.*

   We assumed that the sequence $\{\boldsymbol{x}_k\}$ is the ouput of a tap-delay line with independent inputs, and that the noise sequence $\{v(k)\}$ is independent of $\{\boldsymbol{x}_k\}$ (this last assumption holds in the case of Gaussian inputs and noise). Relaxing these assumptions would require the study of even larger matrices than treated in Chapter 4. Nevertheless, techniques similar to the ones employed here can still be used — the matrix Φ will still have considerable structure.

3. *Exploring the relation between singular perturbation and averaging/ ODE methods.*

   Singular perturbation techniques are widely used in control applications, mostly for continuous-time applications (discrete-time singular perturbation theory is not as well developed). We can show (for filter order $M = 2$,

see Appendix 4.H), that the averaged systems obtained from the ODE method corresponds to the slow subsystem (in the language of singular perturbation theory) of the state-space system (4.H.4) described in Appendix. 4.H.

The state-space linear system (4.8) used in Chapter 4 to describe the second-order statistics of $\tilde{\boldsymbol{w}}_k$ in LMS can also be studied using singular perturbation methods. Pursuing this connection may lead to new and simpler stability conditions for adaptive algorithms, and to estimates on the error incurred in the approximations.

4. *Extending the state-space stability analysis to other adaptive algorithms.*

   Presently, the stability analysis described in this chapter applies only to the LMS algorithm, but the procedure is general enough and can be applied in other contexts.

Chapter 5

1. *Extending the learning-curves analysis to other adaptive algorithms.*

   The results of Chapter 5 apply directly only to the LMS algorithm and to the NLMS algorithm (by using the change of variables that was proposed in Chapter 2). Nevertheless, phenomena similar to those described in Chapter 5 should be present in most, if not all, gradient-based adaptive algorithms.

2. *Extending the learning-curves analysis to non-independent input sequences.*

   Results presented in [BA81] discuss methods (using the Ergodic Theorem)

that might help to extend the results of Chapter 5 to more general input sequences.

3. *Exploring the differences between almost-sure and mean-square analyses*

   In several situations where fast convergence is of interest, the interplay between mean-square and almost-sure convergence and performance results should be considered. In fact, in applications for which the initial condition $\boldsymbol{w}_0$ is good, almost-sure rates of convergence should play no role (since they are manifested only for large time instants k). On the other hand, in situations where the initial condition may be far from the Wiener solution, the faster almost-sure rate of convergence should be important and taken into account for the design of the filter.

Chapter 6

1. *Allowing for a discontinuous α_c in the update* (6.22) *for circular leaky.*

   The leakage function $\alpha_c(\cdot)$ described in Chapter 6 (Eq. (6.23)) was chosen to be differentiable, so that results from averaging theory in the literature could be applied. However, simulations showed that a simpler, discontinuous update equation would lead to a simpler algorithm with essentially the same properties. We studied this function in [NS96] in a deterministic setting. A stochastic analysis would be desirable as well.

2. *Different leakage strategies.*

   The circular-leaky algorithm, although less costly, has more stringent stability conditions than the switching-σ algorithm. An algorithm that is more

robust than circular-leaky, and yet not as costly as switching-σ (depending on the implementation) was proposed in [NS96]. Its performance was analyzed in that reference from a purely deterministic point of view. A stochastic analysis would also be of interest.

Feedback analysis

An alternative approach to the analysis of adaptive filters in terms of a feedback structure with a lossless feedforward path, was recently proposed in [SR96, RS96], and later extended to stochastic analysis in [Mai98, MS99, YS].

Using the feedback structure, references [MS99, YS] show that several steady-state performance results (old and new) can be obtained with considerable less effort than existing derivations.

It is therefore desirable to investigate the implication of this new approach to the study of both the stability and performance of adaptive filters for both slow and non-slow adaptations.

Robust estimation schemes

A new family of estimation problems was recently proposed and solved in [CGG98, SNC98]. This approach accounts for errors not only in the observed vector ($y(k)$ in the notation used in this dissertation), but also in the data used for identification ($\boldsymbol{x}_k$). Several different cost functions have been proposed, and applications in areas such as image restoration, regularization, and robust control [SNC98, NS99b, SN99] have been successfully pursued. A preliminary adaptive version was developed in [SGN97, SGC97]. Its behavior, however, is still not fully understood. Moreover, the analyses so far have been entirely deterministic and stochastic analyses would be desirable as well, in addition to new adaptive variants.

References

[ABH97] P.E. An, M. Brown, and C.J. Harris. "On the convergence rate performance of the normalized least-mean-square adaptation." *IEEE Transactions on Neural Networks*, **8**(5):1211–1214, September 1997.

[Ale87] S.T. Alexander. "Transient weight misadjustment properties for the finite precision LMS algorithm." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-35**(9):1250–1258, September 1987.

[BA80a] R.R. Bitmead and B.D.O. Anderson. "Lyapunov techniques for the exponential stability of linear difference equations with random coefficients." *IEEE Transactions on Automatic Control*, **AC-25**:782–787, 1980.

[BA80b] R.R. Bitmead and B.D.O. Anderson. "Performance of adaptive estimation algorithms in dependent random environments." *IEEE Transactions on Automatic Control*, **AC-25**:788–794, 1980.

[BA81] R.R. Bitmead and B.D.O. Anderson. "Adaptive frequency sampling filters." *IEEE Transactions on Circuits and Systems*, **CAS-28**(6):524–533, June 1981.

[BAN86] R.R. Bitmead, B.D.O. Anderson, and T.S. Ng. "Convergence rate determination for gradient-based adaptive estimators." *Automatica*, **22**:185–191, 1986.

[BB96a] J.C.M. Bermudez and N.J. Bershad. "A nonlinear analytical model for the quantized LMS algorithm — The arbitrary step size case." *IEEE Transactions on Signal Processing*, **44**(5):1175–1183, May 1996.

[BB96b] J.C.M. Bermudez and N.J. Bershad. "Transient and tracking performance analysis of the quantized LMS algorithm for time-varying system identification." *IEEE Transactions on Signal Processing*, **44**(8):1990–1997, August 1996.

[Ben87] A. Benveniste. "Design of adaptive algorithms for the tracking of time-varying systems." *International Journal of Adaptive Control and Signal Processing*, pp. 3–27, 1987.

[Ber86] N.J. Bershad. "Analysis of the normalized LMS algorithm with Gaussian inputs." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-34**:793–806, 1986.

[Bit81] R.R. Bitmead. "Convergence properties of LMS adaptive estimates with unbounded dependent inputs." In *Proceedings of the 20th Conference in Decision and Control*, San Diego, CA, 1981.

[Bit83] R.R. Bitmead. "Convergence in distribution of LMS-type adaptive parameter estimates." *IEEE Transactions on Automatic Control*, **AC-28**:54–60, 1983.

[BMP87] A. Benveniste, M. Métivier, and P. Priouret. *Adaptive Algorithms and Stochastic Approximations*. Springer-Verlag, New York, 1987.

[BS97] K.D. Benson and W.A. Sethares. "Magnitude response peak detection and control using balanced model reduction and leakage to a target." *IEEE Transactions on Signal Processing*, **45**(10):2442–2453, 1997.

[CG91] H.F. Chen and L. Guo. *Identification and Stochastic Adaptive Control*. Birkhauser, Boston, MA, 1991.

[CGG98] S. Chandrasekaran, G. Golub, M. Gu, and A.H. Sayed. "Parameter estimation in the presence of bounded data uncertainties." *SIAM J. Matrix Analysis and Applications*, **19**(1):235–252, January 1998.

[Cio87] J.M. Cioffi. "Limited-precision effects in adaptive filtering." *IEEE Transactions on Circuits and Systems*, **CAS-34**(7):821–833, July 1987.

[CL84] C. Caraiscos and B. Liu. "A roundoff error analysis of the LMS adaptive algorithm." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-32**(1):34–41, February 1984.

[CM75] D.L. Cohn and J.L. Melson. "The residual encoder: an improved ADPCM system for speech digitization." *IEEE Transactions on Communications*, **COM-23**:935–941, September 1975.

[Cra46] H. Cramér. *Mathematical Methods of Statistics*. Princeton University Press, 1946.

[CT91] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 1991.

[CW85] J.M. Cioffi and J.J. Werner. "The tap-drifting problem in digitally implemented data-driven echo cancellers." *Bell Syst. Tech. J.*, **64**(1):115–138, January 1985.

[Dan70] T.P. Daniell. "Adaptive estimation with mutually correlated training sequences." *IEEE Trans. Syst. Sci. Cybern.*, **SSC-6**:12–19, January 1970.

[Dav70] L.D. Davisson. "Steady-state error in adaptive mean-square minimization." *IEEE Transactions on Information Theory*, **IT-16**:382–385, July 1970.

[DP95] S.C. Douglas and W. Pan. "Exact expectation analysis of the LMS adaptive filter." *IEEE Transactions on Signal Processing*, **43**(12):2863–2871, December 1995.

[Dur96] R. Durrett. *Probability: Theory and Examples.* Duxbury Press, 2nd edition, 1996.

[EM85] E. Eweda and O. Macchi. "Tracking error bounds of adaptive nonstationary filtering." *Automatica*, pp. 296–302, 1985.

[Ewe94] E. Eweda. "Comparison of RLS, LMS, and sign algorithms for tracking randomly time-varying channels." *IEEE Transactions on Signal Processing*, **42**(11):2937–2944, November 1994.

[Ewe97] E. Eweda. "A quantitative and tracking comparison of two adaptive echo-cancellation algorithms." *Signal Processing*, **59**:285–290, 1997.

[FB88] J.B. Foley and F.M. Boland. "A note on the convergence analysis of LMS adaptive filters with Gaussian data." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-36**:1087–1089, July 1988.

[FF86] S. Florian and A. Feuer. "Performance analysis of the LMS algorithm with a tapped delay line (two-dimensional case)." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-34**(6):1542–1549, December 1986.

[FW85] A. Feuer and E. Weinstein. "Convergence analysis of LMS filters with uncorrelated Gaussian data." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-33**(1):222–229, February 1985.

[Gan60] F.R. Gantmacher. *The Theory of Matrices.* Chelsea Publishing Company, 1960.

[GHW92] R.D. Gitlin, J.S. Hayes, and S.B. Weinstein. *Data Communications Principles.* Plenum Press, NY, 1992.

[GL89] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins, 2nd edition, 1989.

[GMW82] R.D. Gitlin, H.C. Meadors, Jr., and S.B. Weinstein. "The tap-leakage algorithm: An algorithm for the stable operation of a digitally implemented, fractionally spaced adaptive equalizer." *The Bell System Technical Journal*, **61**(8):1817–1839, October 1982.

[God80] D.N. Godard. "Self-recovering equalization and carrier tracking in two-dimensional data communication systems." *IEEE Transactions on Communications*, **28**(11):1867–1875, November 1980.

[Guo94] L. Guo. "Stability of recursive stochastic tracking algorithms." *SIAM Journal of Control and Optimization*, **32**(5):1195–1225, September 1994.

[Hay96] S. Haykin. *Adaptive Filter Theory.* Prentice-Hall, 3rd edition, 1996.

[HJ87] R.A. Horn and C.R. Johnson. *Matrix Analysis.* Cambridge University Press, 1987.

[HJ94] R.A. Horn and C.R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, Cambridge, MA, 1994.

[Hsi83] T.C. Hsia. "Convergence analysis of LMS and NLMS adaptive algorithms." In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pp. 667–670, Boston, MA, 1983.

[HSK96] B. Hassibi, A.H. Sayed, and T. Kailath. "H^∞ optimality of the LMS algorithm." *IEEE Trans. on Signal Processing*, **44**(2):267–280, February 1996.

[IK84] P.A. Ioannou and P. V. Kokotović. "Instability analysis and improvement of robustness of adaptive control." *Automatica*, **20**(5):583–594, 1984.

[IS96] P.A. Ioannou and J. Sun. *Robust Adaptive Control.* PTR Prentice-Hall, NJ, 1996.

[IT86] P. Ioannou and K. Tsakalis. "Robust discrete-time adaptive control." In K. S. Narendra, editor, *Adaptive and Learning Systems: Theory and Applications*, pp. 73–87. Plenum Press, 1986.

[JCR82] S.K. Jones, R.K. Cavin, and W.M. Reed. "Analysis of error-gradient adaptive linear estimators for a class of stationary dependent processes." *IEEE Transactions on Information Theory*, **IT-28**:318–329, March 1982.

[KA96] W.S. Kafri and E.H. Abed. "Stability analysis of discrete-time singularly perturbed systems." *IEEE Transactions on Automatic Control*, **43**(10):848–850, October 1996.

[Kai80] T. Kailath. *Linear Systems.* Prentice-Hall, Englewood Cliffs, NJ, 1980.

[KBB86] P.V. Kokotović, A. Bensoussan, and G. Blankenship, editors. *Singular Perturbations and Asymptotic Analysis in Control Systems.* Springer-Verlag, 1986.

[KD75] J.-K. Kim and L.D. Davisson. "Adaptive linear estimation for stationary M-dependent processes." *IEEE Transactions on Information Theory*, **IT-21**:23–31, January 1975.

[Kha96] H.K. Khalil. *Nonlinear Systems.* Prentice-Hall, 2nd edition, 1996.

[KKJ86] P.V. Kokotović, H.K. Khalil, and J.O'Reilly. *Singular Perturbation Methods in Control: Analysis and Design.* Academic Press, 1986.

[Koz69] F. Kozin. "A survey of stability of stochastic systems." *Automatica*, **5**:95–112, 1969.

[KSH99] T. Kailath, A.H. Sayed, and B. Hassibi. *State-Space Estimation Theory.* Prentice-Hall, NJ, 1999.

[Kus84] H.J. Kushner. *Approximation and Weak Convergence Methods for Random Processes, with Applications to Stochastic System Theory.* MIT Press, Cambridge, MA, 1984.

[KV96] H.J. Kushner and F.J. Vázquez-Abad. "Stochastic Approximation Methods for systems over an infinite horizon." *SIAM Journal of Control and Optimization*, **34**(2):712–756, March 1996.

[KY97] H.J. Kushner and G.G. Yin. *Stochastic Approximation Algorithms and Applications.* Applications of Mathematics. Springer, 1997.

[Lin88] F. Ling. "Convergence characteristics of LMS and LS adaptive algorithms for signals with rank-deficient correlation matrices." In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 1499–1502, New York, December 1988.

[Lju77] L. Ljung. "Analysis of recursive stochastic algorithms." *IEEE Transactions on Automatic Control*, **22**:551–575, August 1977.

[LL92] T-H.S. Li and J-H. Li. "Stabilization bound of discrete two-time-scale systems." *Systems & Control Letters*, **18**:479–489, 1992.

[MA97] K. Mayyas and T. Aboulnasr. "The leaky LMS algorithm: MSE analysis for Gaussian data." *IEEE Transactions on Signal Processing*, **45**(4):927–934, April 1997.

[Mac95] O. Macchi. *Adaptive Processing — The Least Mean Squares Approach with Applications in Transmission.* Wiley, England, 1995.

[Mai98] J. Mai. *"Steady-State Performance of Fractionally Spaced Blind Adaptive Equalizers."*. Master's thesis, Department of Electrical Engineering, University of California, Los Angeles, 1998.

[Maz79] J.E. Mazo. "On the independence theory of equalizer convergence." *The Bell System Technical Journal*, **58**:963–993, May–June 1979.

[ME83] O. Macchi and E. Eweda. "Second-order convergence analysis of stochastic adaptive linear filtering." *IEEE Transactions on Automatic Control*, **AC-28**(1):76–85, January 1983.

[Mit70] D.S. Mitrinović. *Analytic Inequalities.* Springer-Verlag, 1970.

[MS99] J. Mai and A.H. Sayed. "A feedback approach to the steady-state performance of fractionally-spaced blind adaptive equalizers." Submitted for publication, 1999.

[NS96] V.H. Nascimento and A.H. Sayed. "An unbiased and cost-effective leaky-LMS filter." In *Conference Record of the 30th Asilomar Conference on Signals, Systems, and Computers*, volume 2, pp. 1078–1082, Pacific Grove, CA, November 1996.

[NS98a] V. H. Nascimento and A. H. Sayed. "Stability of the LMS adaptive filter by means of a state equation." In *Proc. Allerton Conference*, Monticello, IL, September 1998.

[NS98b] V.H. Nascimento and A.H. Sayed. "Are Ensemble-Average Learning Curves Reliable in Evaluating the Performance of Adaptive Filters?" In *Proceedings of the 32nd Asilomar Conference on Signals, Systems, and Computers*, pp. 1171–1175, Pacific Grove, CA, November 1998.

[NS99a] V.H. Nascimento and A.H. Sayed. "On the proper interpretation of ensemble-average learning curves for adaptive filtering." Submitted for publication, 1999.

[NS99b] V.H. Nascimento and A.H. Sayed. "Optimal State Regulation for Uncertain State-Space Models." In *Proc. ACC*, San Diego, CA, June 1999.

[NS99c] V.H. Nascimento and A.H. Sayed. "Unbiased and stable leakage-based adaptive filters." To appear in *IEEE Transactions on Signal Processing*, 1999.

[Pap84] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, 2nd edition, 1984.

[PE78] A. Parthasarathy and R.M. Evan-Iwanowski. "On the almost sure stability of linear stochastic systems." *SIAM J. Appl. Math.*, **34**(4):643–656, June 1978.

[Pro89] J.G. Proakis. *Digital Communications.* McGraw Hill, 2nd edition, 1989.

[PTV94] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C.* Cambridge University Press, 2nd edition, 1994.

[QG77] S.U.H. Qureshi and Jr. G.D. Forney. "Performance and properties of a $T/2$ equalizer." In *Conference Record, NTC'1977*, pp. 11:1.1–4, December 1977.

[Rao81] C.V.K. Prabhakara Rao. "Analysis of the steady-state performance in adaptive echo cancelers with correlated data and correlated received signal." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-29**(6):1222–1225, December 1981.

[RBJ91] G.J. Rey, R.R. Bitmead, and C.R. Johnson, Jr. "The Dynamics of bursting in simple adaptive feedback Systems with Leakage." *IEEE Transactions on Circuits and Systems*, **CAS-38**(5):475–488, May 1991.

[Red93] D. Redfern. *Maple Handbook.* Spring Verlag, 1993.

[RM51] H. Robbins and S. Monro. "A stochastic approximation method." *Annals of Mathematical Statistics*, **22**:400–407, 1951.

[RS96] M. Rupp and A.H. Sayed. "A time-domain feedback analysis of filtered-error adaptive gradient algorithms." *IEEE Transactions on Signal Processing*, **44**:1428–1439, 1996.

[Rud76] W. Rudin. *Principles of Mathematical Analysis.* McGraw-Hill, 3rd edition, 1976.

[Rup93] M. Rupp. "The behavior of LMS and NLMS algorithms in the presence of spherically invariant processes." *IEEE Transactions on Signal Processing*, **SP-41**:1149–1160, March 1993.

[Rup95] M. Rupp. "Bursting in the LMS algorithm." *IEEE Transactions on Signal Processing*, **43**(10):2414–2417, October 1995.

[Say97] A.H. Sayed. "Lecture Notes on Adaptive Filtering." UCLA Dept. of Electrical Engineering, 1997.

[Set92] W.A. Sethares. "Adaptive algorithms with nonlinear data and error functions." *IEEE Transactions on Signal Processing*, **40**(9):2199–2206, September 1992.

[Set93] W.A. Sethares. "The least mean square family." In N. Kalouptsidis and S. Theodoridis, editors, *Adaptive System Identification and Signal Processing Algorithms.* Prentice-Hall, 1993.

[SGC97] A.H. Sayed, A. Garulli, and S. Chandrasekaran. "A fast iterative solution for worst-case parameter estimation with bounded model uncertainties." In *Proceedings of the 1997 American Control Conference*, volume 3, pp. 1499–1503, Albuquerque, NM, June 1997.

[SGN97] A.H. Sayed, A. Garulli, V.H. Nascimento, and S. Chandrasekaran. "Exponentially-weighted iterative solutions for worst-case parameter estimation." In *Proc. 5th IEEE Mediterranean Conference on Control and Systems*, Cyprus, July 1997.

[SK94] A.H. Sayed and T. Kailath. "A state-space approach to adaptive RLS filtering." *IEEE Signal Processing Magazine*, **11**(3):18–60, July 1994.

[SK95] V. Solo and X. Kong. *Adaptive Signal Processing Algorithms.* Prentice-Hall, Englewood Cliffs, NJ, 1995.

[SLJ86] W.A. Sethares, D.A. Lawrence, C.R. Johnson, Jr., and R.R. Bitmead. "Parameter drift in LMS adaptive filters." *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **ASSP-34**:868–878, August 1986.

[Slo93] D.T.M. Slock. "On the convergence behavior of the LMS and the normalized LMS algorithms." *IEEE Transactions on Signal Processing*, **41**(9):2811–2825, September 1993.

[SN99] A.H. Sayed and V.H. Nascimento. "Design Criteria for Uncertain Models with Structured and Unstructured Uncertainties." To appear in *Robustness in Identification and Control*, A. Garulli, A. Tesi, and A. Vicino, eds. Lecture Notes in Control and Information Sciences., 1999.

[SNC98] A.H. Sayed, V.H. Nascimento, and S. Chandrasekaran. "Estimation and control with bounded data uncertainties." *Linear Algebra and Its Applications*, pp. 259–306, December 1998.

[SOK84] V.R. Saksena, J. O'Reilly, and P.V. Kokotović. "Singular perturbations and time-scale methods in control theory: survey 1976–83." *Automatica*, **20**(3):273–293, May 1984.

[Sol89] V. Solo. "The limiting behavior of LMS." *IEEE Transactions on Signal Processing*, **37**(12):1909–1922, December 1989.

[Sol92] V. Solo. "The error variance of LMS with time-varying weights." *IEEE Transactions on Signal Processing*, **40**(4):803–813, April 1992.

[Sol94a] V. Solo. "Averaging analysis of the LMS algorithm." In C.T. Leondes, editor, *Control and Dynamic Systems*, volume 65: Stochastic Techniques in Digital Signal Processing Systems, Part 2 of 2, pp. 379–397. Academic Press, 1994.

[Sol94b] V. Solo. "On the stability of slowly time-varying linear systems." *Mathematics of control, Signals, and Systems*, **7**:331–350, 1994.

[Sol97] V. Solo. "The stability of LMS." *IEEE Transactions on Signal Processing*, **45**(12):3017–3026, December 1997.

[SR96] A.H. Sayed and M. Rupp. "Error-energy bounds for adaptive gradient algorithms." *IEEE Transactions on Signal Processing*, **44**(8):1982–1989, August 1996.

[SR98] A.H. Sayed and M. Rupp. "Robustness issues in adaptive filtering." In *DSP Handbook*, chapter 20. CRC Press, 1998.

[SSB96] R. Sharma, W.A. Sethares, and J.A. Buclew. "Asymptotic analysis of stochastic gradient-based adaptive filtering algorithms with general cost functions." *IEEE Transactions on Signal Processing*, **44**(9):2186–2194, September 1996.

[Ste73] G.W. Stewart. *Introduction to Matrix Computations.* Academic Press, 1973.

[TC96] O. Tanrikulu and J.A. Chambers. "Convergence and steady-state properties of the least-mean mixed-norm (LMMN) adaptive algorithm." *IEEE Transactions on Signal Processing*, **143**(3):137–142, June 1996.

[TF88] M. Tarrab and A. Feuer. "Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data." *IEEE Transactions on Information Theory*, **34**(4):680–691, July 1988.

[TFC96] J.R. Treichler, I. Fijalkow, and Jr. C.R. Johnson. "Fractionally spaced equalizers — how long should they really be?" *IEEE Signal Processing Magazine*, pp. 65–81, May 1996.

[TK82] D.W. Tufts and R. Kumaresan. "Estimation of frequencies of multiple sinusoids: making linear prediction perform like maximum likelihood." *Proceedings of the IEEE*, **70**:975–989, September 1982.

[Ung74] G. Ungerboeck. "Fractional tap-spacing equalizer and consequences for clock recovery in data modems." *IEEE Transactions on Communications*, **COM-24**(8):856–864, August 1974.

[VVK94] S. Vembu, S. Verdú, R.A. Kennedy, and W. Sethares. "Convex cost functions in blind equalization." *IEEE Transactions on Signal Processing*, **42**(8):1952–1960, August 1994.

[Wil63] J.H. Wilkinson. *Rounding Errors in Algebraic Processes.* Her Majesty's Stationery Office, 1963.

[WJM75] B. Widrow, Jr. J.R. Glover, J.M. McCool, J. Kaunitz, C.S. Williams, R.H. Hearn, J.R. Zeidler, Jr. E. Dong, and R.C. Goodlin. "Adaptive noise cancelling: principles and applications." *Proceedings of the IEEE*, **63**(12):1692–1716, December 1975.

[WL90] B. Widrow and M.A. Lehr. "30 years of adaptive neural networks: perceptron, madaline, and backpropagation." *Proceedings of the IEEE*, **78**(9):1415–1441, 1990.

[WM79] A. Weiss and D. Mitra. "Digital adaptive filters: conditions for convergence, rates of convergence, effects of noise and errors arising from the implementation." *IEEE Transactions on Information Theory*, **IT-25**(6):637–652, November 1979.

[WMG67] B. Widrow, P. Mantey, L. Griffiths, and B. Goode. "Adaptive antenna systems." *Proceedings of the IEEE*, **55**:2143–2159, December 1967.

[WML76] B. Widrow, J.M. McCool, M.G. Larimore, and Jr. C.R. Johnson. "Stationary and nonstationary learning characteristics of the LMS adaptive filter." *Proceedings of the IEEE*, **64**(8):1151–1162, August 1976.

[WS85] B. Widrow and S.D. Stearns. *Adaptive Signal Processing.* Prentice-Hall, Englewood Cliffs, 1985.

[WS90] G.A. Williamson and W.A. Sethares. "Robustness to quantization errors in LMS adaptation via degree of excitation." In *Proceedings IEEE 1990 ICASSP*, pp. 1483–1486, April 1990.

[WW84] E. Walach and B. Widrow. "The least mean fourth (LMF) adaptive algorithm and its family." *IEEE Transactions on Information Theory*, **IT-30**(2):275–283, March 1984.

[YS] N. Yousef and A. H. Sayed. "A unified approach to the steady-state performance analysis of stochastic gradient adaptive algorithms for both stationary and nonstationary environments." In preparation.

[Zah73] C.L. Zahm. "Application of adaptive arrays to suppress strong jammers in the presence of weak signals." *IEEE Transactions on Aerospace and Electronic Systems*, **AES-9**(2):260–271, March 1973.